AD-A043 048   E-SYSTEMS INC FALLS CHURCH VA MELPAR DIV   F/G 9/5
APPLICATION OF MICROPROCESSORS FOR MINIATURIZED TMDE.(U)
JUN 77 F E LANHAM   DAAB07-76-C-1279
UNCLASSIFIED   5073   ECOM-76-1279-F   NL

1 OF 1
ADA043048

END
DATE
FILMED
9-77
DDC

Research and Development Technical Report
ECOM-76-1279-F

12

APPLICATION OF MICROPROCESSORS FOR MINIATURIZED TMDE

Frank E. Lanham
E-Systems, Inc., Melpar Division
7700 Arlington Boulevard
Falls Church, Virginia 22046

June 1977

Final Report for Period 23 July 1975 to 23 December 1976

DISTRIBUTION STATEMENT

Approved for public release;
distribution unlimited.

Prepared for:

ECOM

US ARMY ELECTRONICS COMMAND FORT MONMOUTH, NEW JERSEY 07703

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The citation of trade names and names of manufacturers in this report is not to be construed as official Government indorsement or approval of commercial products or services referenced herein.

Disposition

Destroy this report when it is no longer needed. Do not return it to the originator.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: 18 ECOM-76-1279-F	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): 6 Application of Microprocessors for Miniaturized TMDE.	5. TYPE OF REPORT & PERIOD COVERED: 9 FINAL Report. 23 July 1975 - 23 Dec 1976,
	6. PERFORMING ORG. REPORT NUMBER: 14 5073
7. AUTHOR(s): 10 Frank E. Lanham	8. CONTRACT OR GRANT NUMBER(s): 15 DAAB07-76-C-1279 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS: E-Systems, Inc., Melpar Division, 7700 Arlington Boulevard, Falls Church, Virginia 22046	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 16 1L7 62705 AH94 T1; 17 T1
11. CONTROLLING OFFICE NAME AND ADDRESS: U. S. Army Electronics Command, ATTN: DRSEL-TL-MI, Fort Monmouth, New Jersey 07703	12. REPORT DATE: 11 June 1977
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): N/A 12 64p.	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release;
distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

N/A

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Test, Measurement and Diagnostic Equipment
Automatic Test Equipment
Maintenance
Maintenance Equipment
Test Equipment
Ground Support Equipment

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The goal of this project was to establish the feasibility of developing miniaturized, automatic contact and repair test equipment (CARTE) using a microprocessor as the control and computational element. By changing the firmware via either cassette tapes or plug-in ROM's, the CARTE could be used by contact teams in the field to diagnose and repair a variety of operational equipment. The use of third generation automatic test equipment (ATE) techniques was to be emphasized in order to minimize

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

407 698

LB

the amount of hardware required. A laboratory demonstration system was assembled to provide basic stimulus, measurement, and testing capabilities based on a microprocessor and using third generation ATE techniques. The system hardware consisted of the microprocessor and its memory and I/O bus, a digitally synthesized software controlled stimulus generator, a software controlled amplitude sampling unit, a frequency counter under remote control via an IEEE bus, and various peripherals. The peripherals included two cassette transports for data storage, a special function keyboard, a CRT terminal, and a serial printer for hardcopy of the CRT screen contents. The special hardware designs of the digitally synthesized stimulus generator and amplitude sampling units permit the microprocessor software to control both the setup and operation of both the stimulus and measurement functions of the laboratory model. The software written for the system included a multi-tasking operating system, a set of computational software which implemented stimulus and measurement capabilities through the use of third generation ATE techniques, and a top level demonstration program which made all system hardware and software resources available to the system user by means of mnemonic commands entered via the CRT terminal keyboard. The demonstration program additionally implemented an automated gain test for audio amplifiers using resources supplied by the lower level computational software. The lab system convincingly demonstrated the feasibility of using a microprocessor as the basis for CARTE. The concepts and techniques developed on this exploratory development program can be improved and expanded and implemented in a small, field deployable unit.

SUMMARY

The goal of this project was to establish the feasibility of developing miniaturized, automatic contact and repair test equipment (CARTE) using a microprocessor as the control and computational element. By changing the firmware via either cassette tapes or plug-in ROM's, the CARTE could be used by contact teams in the field to diagnose and repair a variety of operational equipment. The use of third generation automatic test equipment (ATE) techniques was to be emphasized in order to minimize the amount of hardware required.

A laboratory demonstration system was assembled to provide basic stimulus, measurement, and testing capabilities based on a microprocessor and using third generation ATE techniques. The system hardware consisted of the microprocessor and its memory and I/O bus, a digitally synthesized software controlled stimulus generator, a software controlled amplitude sampling unit, a frequency counter under remote control via an IEEE bus, and various peripherals. The peripherals included two cassette transports for data storage, a special function keyboard, a CRT terminal, and a serial printer for hardcopy of the CRT screen contents. The special hardware designs of the digitally synthesized stimulus generator and amplitude sampling units permit the microprocessor software to control both the setup and operation of both the stimulus and measurement functions of the laboratory model. The software written for the system included a multitasking operating system, a set of computational software which implemented stimulus and measurement capabilities through the use of third generation ATE techniques, and a top level demonstration program which made all system hardware and software resources available to the system user by means of mnemonic commands entered via the CRT terminal keyboard. The demonstration program additionally implemented an automated gain test for audio amplifiers using resources supplied by the lower level computational software.

The lab system convincingly demonstrated the feasibility of using a microprocessor as the basis for CARTE. The concepts and techniques developed on this exploratory development program can be improved and expanded and implemented in a small, field deployable unit.

It is recommended that a brassboard model be developed utilizing the concept of a basic hardware main frame into which modules required for a given maintenance task can be inserted. The hardware and software should be developed in a form which closely resembles the ultimate production unit.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

INTRODUCTION

The basic objective of this project was to demonstrate the feasibility of using a microprocessor to implement a physically small third generation ATE system.

"Third Generation" ATE

In order to understand what is meant by "third generation" ATE, we will first define "second generation" ATE and then examine the evolution to third generation systems. In second generation ATE the computer acts more or less as a taskmaster and bookkeeper and does not get involved in the actual stimulus and measurement processes necessary for generalized testing. Instead, the computer sends out commands to various specialized stimulus and measurement hardware subsystems, each of which then performs one or more related functions in total upon receiving the computer command. In this type of system, there is a different stimulus subsystem, or "black box," for each type or classification of signal that may have to be generated by the system. In turn, there is generally a separate measurement box for each class of signal measurement the system is required to make. Each of these subsystems is self-sufficient to the extent that it contains all the hardware necessary to make the complete measurements that are requested by the computer. The organization of such a system may be thought of as a star network with the computer at the center connected by lines to each of the stimulus and measurement boxes surrounding it. Figure 1 illustrates this hardware architecture. This second generation approach is also sometimes referred to as the "rack-and-stack" approach because of the large quantities of rack-mounted hardware often required to implement a system of this type.

The shift from second generation to third generation ATE has come about by transferring a large amount of the computational workload that was once performed in the specialized hardware subystems back into the computer. This then allows the multitude of special purpose stimulus and measurement subsystems to be replaced, ideally, with a single stimulus subsystem and a single measurement subsystem, both of which are under detailed computer control. Figure 2 illustrates the simplified hardware architecture resulting from this approach. In a system of this type, the computer sends to the stimulus subsystem in a generalized format a detailed description of the waveform to be generated. Conversely, the measurement subsystem takes many samples of the signal being analyzed and inputs them to the computer, where the raw data is messaged to produce final measurements. These are, in brief,

84671

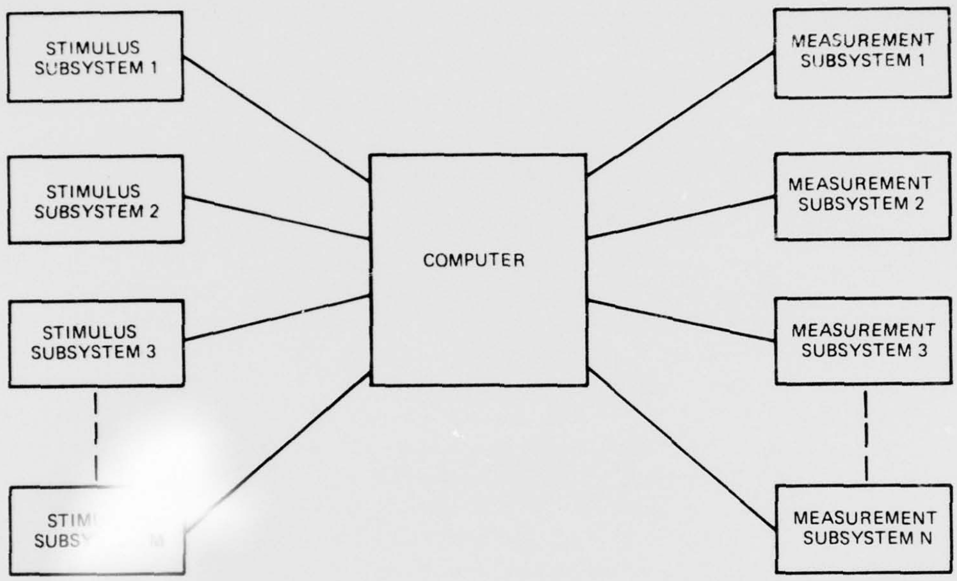

Figure 1. Second Generation ATE Hardware Architecture

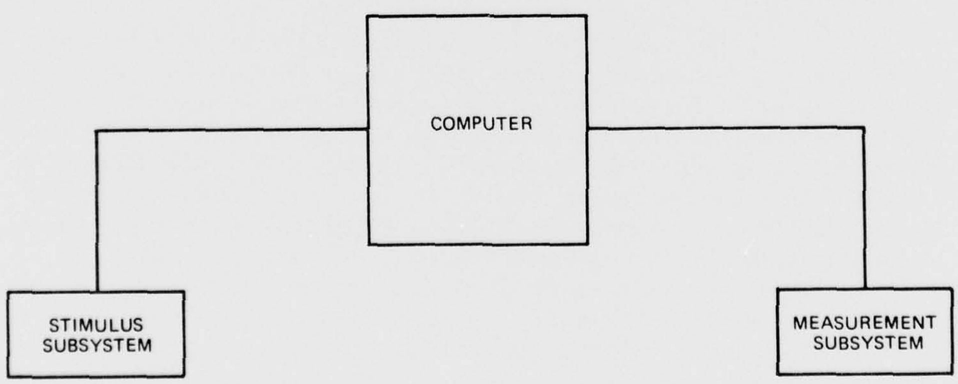

Figure 2. Third Generation ATE Hardware Architecture

the concepts behind the third generation approach to ATE systems. The major benefits realized include reduction of system cost on a recurring production basis and a reduction in system size, weight, and complexity with attendant increases in reliability and portability.

Operational Need/Usage

The Army today finds itself with an extremely large maintenance workload resulting from the need to support its complex weapons systems, communications systems, and other equipments. In order to shoulder this load, it currently maintains an inventory of a multitude of different types of electronic test equipment, some of which are outdated to the point of still using vacuum tube technology. As the Army continues to upgrade this test equipment complement, it would be desirable to also streamline its maintenance operations by upgrading in a manner that results in fewer types of test equipment. A third generation ATE approach is clearly responsive to this aspect of systems maintenance since, by its very nature, it makes a small amount of hardware applicable to a large number of maintenance applications. This is borne out in fact by the Army's successful and growing use of a recently developed third generation ATE system. This system, known as EQUATE, AN/USM-410( )(V), is based on a minicomputer and disk subsystem and is housed in its own air conditioned van for operational use in the field.

In order to satisfy the further requirement for supporting equipment at or near a battle front, a different class of test equipment, sometimes referred to as contact and repair test equipment (CARTE), is required. It is different from a system such as EQUATE in that it must be able to be taken to the field by a "contact team" to repair equipment that cannot be conveniently replaced or sent to the rear for repair. Such a system would, of necessity, be a smaller scale system with more limited capabilities than those systems operating in protected environments with extensive peripherals that are not required to be highly mobile. Ideally, such a piece of test equipment should be able to be carried by one or two men. Thus, it could be loaded onto the back of a jeep, driven to where it is needed and be unloaded and carried by a serviceman to the failed equipment such as, for instance, a tank with a communications subsystem which is suspected of operating improperly. A module containing the test program for the communications equipment in the tank would be plugged into the CARTE, effectively configuring it as a test system targeted directly at the specific equipment in question. With these CARTE resources at his disposal, the maintenance technician would first determine if the subsystem is actually malfunctioning or not. If so, he would continue using the CARTE in an

interactive mode to quickly isolate the fault location so that the failed Line Replaceable Unit (LRU), also identified as Lowest Replaceable Unit, could be replaced to effect immediate repair. The faulty LRU could then be removed to a battalion-level maintenance area where the CARTE would be used to troubleshoot it to a pluggable part level, i.e., to the failed module, printed circuit board, etc., in order to restore a full LRU spares level. At that point, the maintenance technician might or might not attempt to repair the faulty pluggable part. His decision to do so would depend primarily on the complexity of the part. Those pluggable parts not repairable at this level but worth being repaired would be sent back to the rear to be repaired either by traditional manual techniques or by the use of a large-scale ATE system, such as EQUATE.

State of the Art

Until just a few years ago, it would have been either highly impractical or impossible to produce a piece of test equipment of the nature just described. The development of third generation ATE techniques have gone a long way toward making this goal possible: witness the shrinkage of many-rack ATE systems to the point where an equivalent system can be shoehorned into a small van and put in the field. However, the factor that finally makes practical the building of versatile, highly portable ATE systems for contact-type applications is the rapid advance of electronics technology on many fronts simultaneously.

The impact of this advance is felt most heavily in the area of semiconductor devices, primarily microprocessors and memories. Today there exists a spectrum of large scale integration (LSI) microprocessors on a single chip that range from 4-bit machines all the way to 16-bit machines with architectures and instruction sets comparable to those of minicomputers. At this point in time the main area that suffers from the use of microprocessors is processing speed; however, in many computer applications speed is not a constraining factor. The area that gains from the use of microprocessors is the reduction in size, weight, hardware complexity, and power consumption. The development of very large capacity semiconductor memory chips, both read/write memories (RAM) and read-only memories (ROM), is the fulfillment of the dual requirements for processing power and data storage that results in the exceptional amount of capability in a small package needed for a CARTE device of the type being considered here. That is to say, the ability to store sufficient amounts of software in physically small ROM chips (often called firmware) adequate to perform the more limited tasks of a CARTE device

totally frees that device from having to carry with it the usual encumbering computer peripherals used to store and load voluminous amounts of programs and data. The same reductions in size, weight, and power gained from using microprocessors apply equally well but on a larger scale to semiconductor memories. In addition, their performance is actually greater than that of the core memory technology they replace.

Advances in many other areas also contribute to the growing attractiveness of the CARTE approach for field maintenance requirements. For instance, flat panel alphanumeric displays now exist which enhance the desirable interactive man/machine interface capabilities of a system while consuming relatively small amounts of space. The emergence of small, lightweight, and highly efficient switching-regulated power supplies is today a large impetus toward the miniaturization of portable equipment of all types. New packaging and cabling techniques, such as micro-min hybridization techniques and flat ribbon cable, also lend themselves to making high-density systems both reliable and economical. Doubtless further developments in these directions and others will be forthcoming in the future.

Project Objectives and Approach

The efforts of this project have been focused on the primary goal of determining the feasibility of exploiting the microprocessor as the control and computational element in a CARTE type third generation ATE system. Since funding for this work was limited, it was felt that maximum new information would be gained by investing a majority of available resources in software development at the expense of reinventing the hardware wheel. That is, the hardware capability for a system of this type already exists in one form or another and, while it may need modification, does not represent the order of magnitude of lack of knowledge and experience that microprocessor usage does. In the final analysis, the question of microprocessor feasibility is this: Are there any insurmountable problems in the designing, writing, debugging, or execution of the microprocessor software required to implement the desired end system? Thus, the hardware for this project was cast in the form of a fairly minimal laboratory development model limited to analog stimulus and measurement capabilities in order to provide sufficient remaining project resources to pursue this critical microprocessor feasibility/software development question to an informed conclusion.

There was determined to be a basic set of features and capabilities that should be included in the lab model in order to credibly demonstrate the

feasibility of producing a physically small third generation ATE system based on a microprocessor. There should be the ability to digitally synthesize, for stimulus purposes, simple and complex waveforms with variable parameters, such as amplitude, offset, and frequency up to 3 MHz.

During the course of the contract it was agreed that the waveform generation would be limited to several predetermined waveshapes. (See Table 1 on page 34.) Also measurement hardware should be provided to digitize analog samples to 12 bits at a 500 kHz rate. A frequency counter function should be included and should be driven from the IEEE STD-488 digital data bus.

The counter should be present for purposes of providing frequency measurements and for automatically determining the proper sampling rate for any signal having measurements made on it. Automatic signal switching hardware for both stimulus and measurement was considered but rejected as being too costly for the scope of this project. A set of basic measurements should be provided for general purpose use, including such items as average dc value, average ac value, peak-to-peak amplitude, and so forth. There should also be a specific test sequence provided to illustrate the general nature of how testing might be accomplished with this type of system. All of the system's capabilities should be made available to the system user in a convenient, interactive mode using a mnemonic command structure for input and well formatted, decimal displays for output.

Finally, it was decided that this system should be provided with a small scale operating system which, although being general purpose in what it does, is limited in its scope to the needs of this type system. This decision was made in recognition of the fact that an operating system will become invaluable as the system grows and has increasingly complex demands placed upon it. In fact, the exclusion of an operating system of some sort would probably have diluted significantly the value of any positive conclusions regarding the feasibility of microprocessor utilization resulting from this project.

SYSTEM DESCRIPTION

Laboratory Model Approach

In order to keep hardware costs to a minimum on this project, the system was built as a laboratory model and efforts were made to cut costs wherever doing so would not interfere radically with the system's technical quality. The most important cost reduction feature of the lab model approach to system construction is the deletion of packaging of the various system components into a unified structure as would be done for a brassboard model or a prototype. Instead, the different pieces of hardware were arranged on lab workbenches and tables. Other cost cutting items included building cables only to minimum functional requirements and using Melpar-owned lab power supplies instead of buying power supplies for the equipment. The only exception to this was purchased power supplies for the electrically isolated portion of the signal measurement hardware. Instead of designing and building memory cards and a microprocessor card, their function was performed by the purchased microprocessor development system, which contains both the microprocessor and 12,288 sixteen-bit words of semiconductor RAM memory. Melpar provided an ASR-33 Teletype terminal with paper tape reader and punch for use to communicate with the assembler program furnished with the development system.

Rather than incur the expense of designing a custom printed circuit board for each group of analog circuitry, standard PC boards were built with a ground plane on one side, bare fiberglass on the other side, and a pattern to mate with card-edge connectors at the bottom end. A simple tool was made which allowed isolated pads to be manually fabricated on the ground plane side for attachment of component leads. Components were then wired in a point-to-point fashion. All peripherals in the system were bought as off-the-shelf units, with no attempt being made to select units on the basis of size, weight, or ruggedness for inclusion in a prototype or production system. And finally, no finished drawings or wire lists were produced. All work was done from engineering sketches.

The block diagram in figure 3 shows the various peripherals and custom designed hardware circuits and interfaces described in this section. It also illustrates their connection to the PACE microprocessor via the microprocessor bus and interrupt lines.

Figure 4 is a photograph of the total TMDE lab model, including several non-deliverable power supplies and teletypewriter. On the top shelf starting on

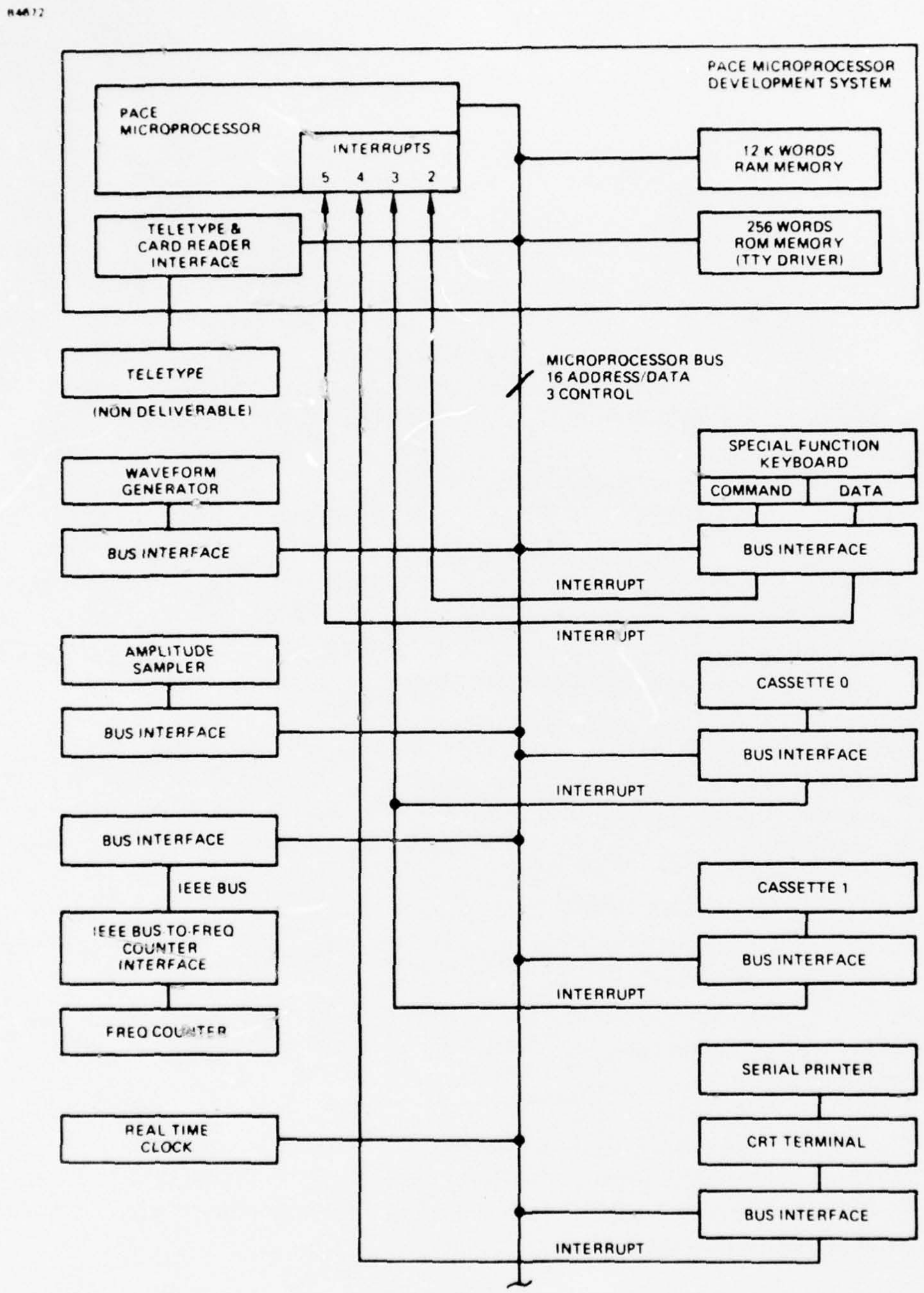

Figure 3. TMDE Lab Model Hardware Block Diagram

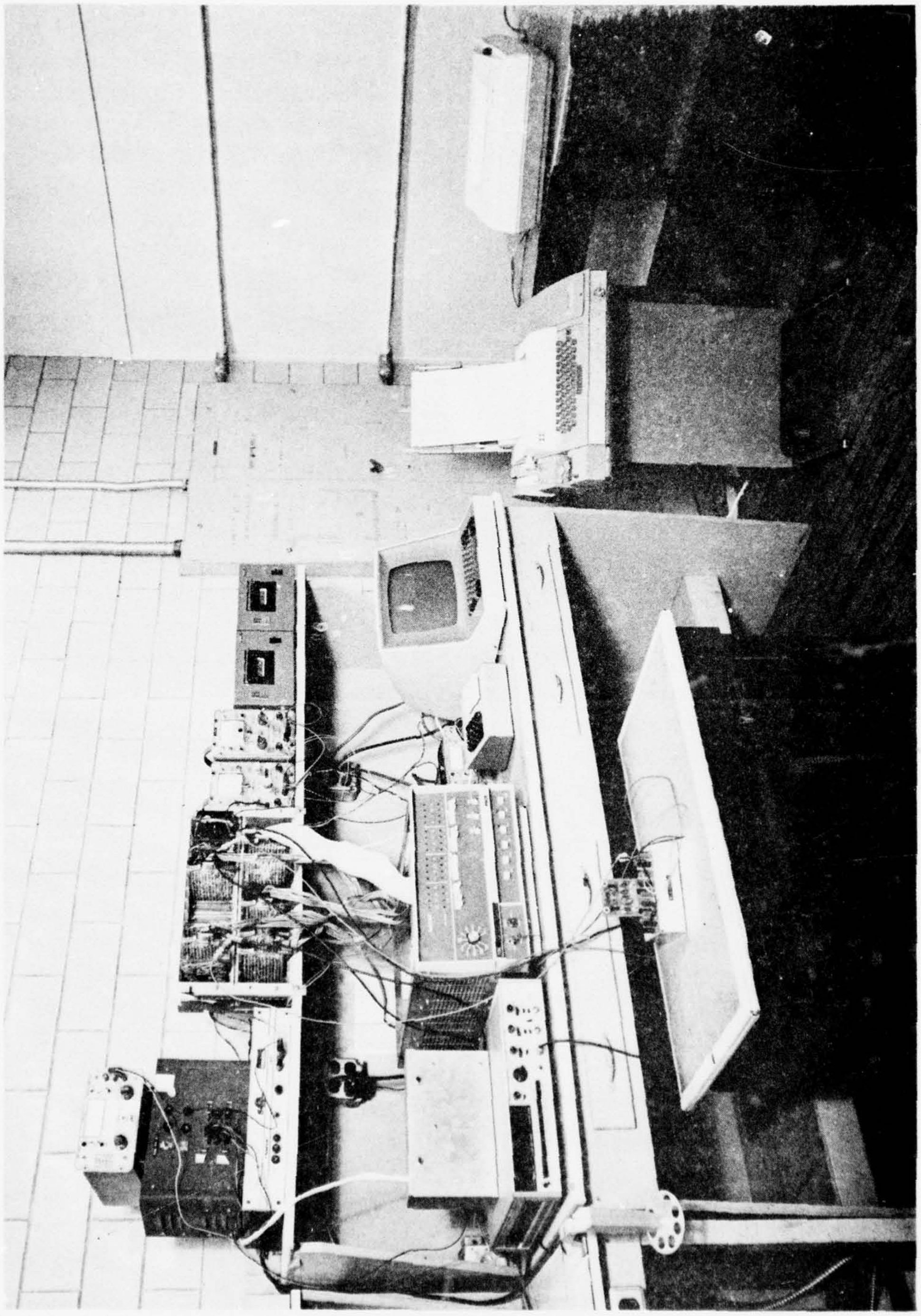

Figure 4. Total TMDE Lab Model

the left are three Melpar lab power supplies, two card cages bolted together containing most of the TMDE model circuits, two more Melpar power supplies, and two digital cassette tape transports made by Ross Controls Corporation. Ross has since been bought and integrated into the Memodyne Corporation. On the lower level at the left is the Systron Donner Model 6150 Counter/Timer topped by the TMDE model interface to the IEEE bus (IEEE Standard 488-1975). Next is the PACE microprocessor development system, built by National Semiconductor Corporation. To the right of it is the TMDE model special function keyboard. Behind it is a Melpar lab power supply. At the right end is the Lear Siegler ADM-1 CRT display terminal. To the right of the workbench is the Melpar Teletype. Next to it is the ISM-80 serial printer from International Systems Marketing. On the small table in front of the workbench is a Melpar audio amplifier board in a test fixture. It was used as a unit under test (UUT) for demonstrating the system. Figure 5 is a photograph of the hardware, without cables, that was actually delivered on this project.

As was noted earlier, the PACE microprocessor development system (MDS) was used to substitute for a processor board and memory boards in the lab model. Figure 6 is a photograph of the PACE MDS, showing its operator's panel. This unit has a cable coming out with a paddle board on the end which makes available all of the PACE bus signals. In the lab model, this paddle board is plugged into a slot in the lower card cage of the double card cage assembly shown in figure 7. This card cage has had its card connectors wired pin-for-pin, thus giving each card slot access to the PACE bus whenever the paddle board is inserted in a slot. Because the backplane has been wired as a bus for both signals and power, the wirewrapped interface cards that fill this card cage are not restricted to any one slot. Each of these interface cards was designed as part of this project.

The top card cage in figure 7 contains the actual stimulus and measurement hardware built during this project. The three cards on the left constitute the waveform generator hardware, including its frequency synthesizer. The four cards on the right make up what is referred to in this report as the amplitude sampler. The rightmost card of this group contains a floating power supply which enables the analog sampling circuits to be electrically isolated from the rest of the system, most of which is digital in nature. The card slots in the top card cage have by necessity been wired in a dedicated fashion, requiring each card to be plugged into a particular slot.

Figure 8 shows typical examples of each of the three types of cards built for the lab model. The card on the left is one of the analog cards from

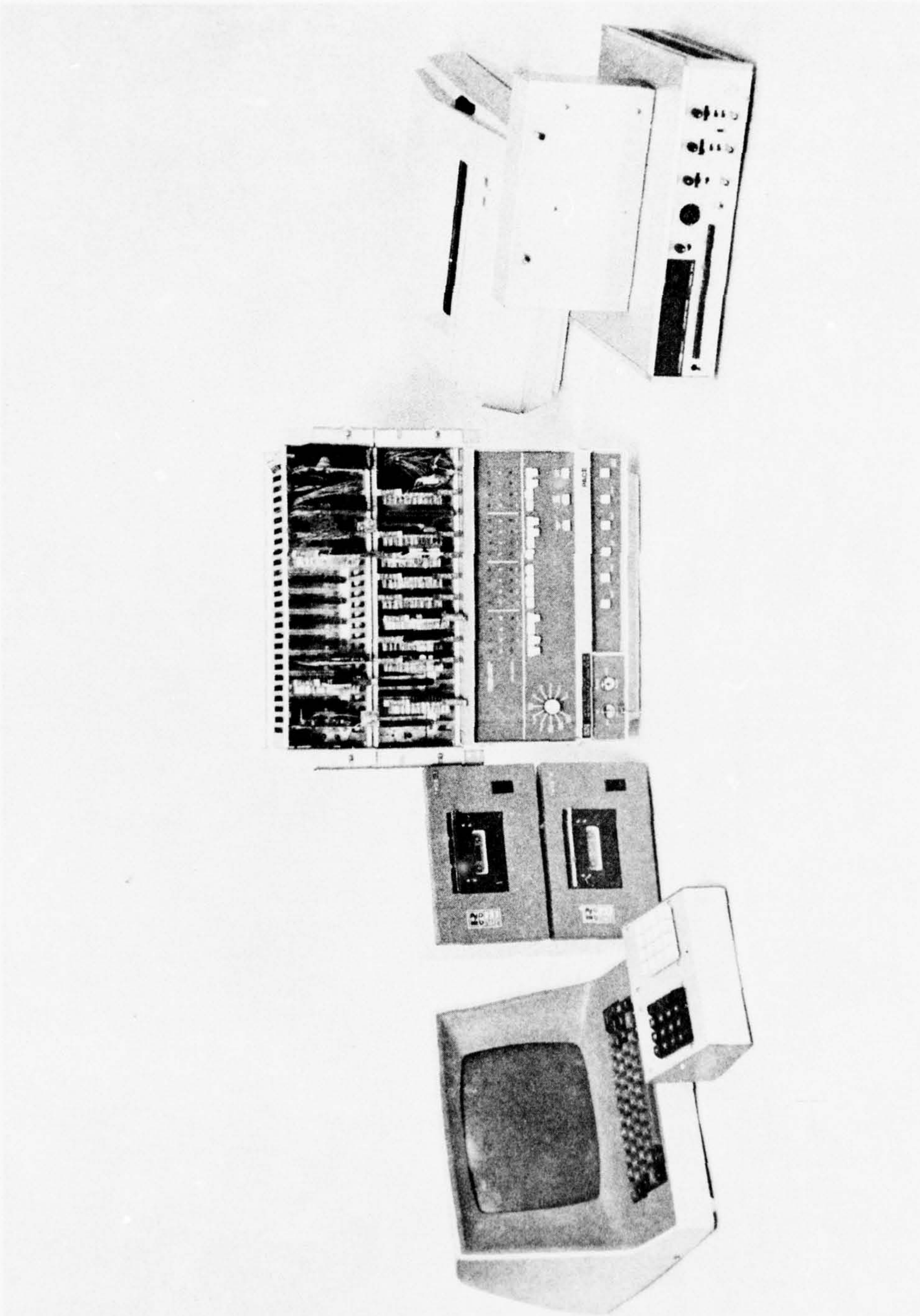

Figure 5. Photograph of Hardware for Project

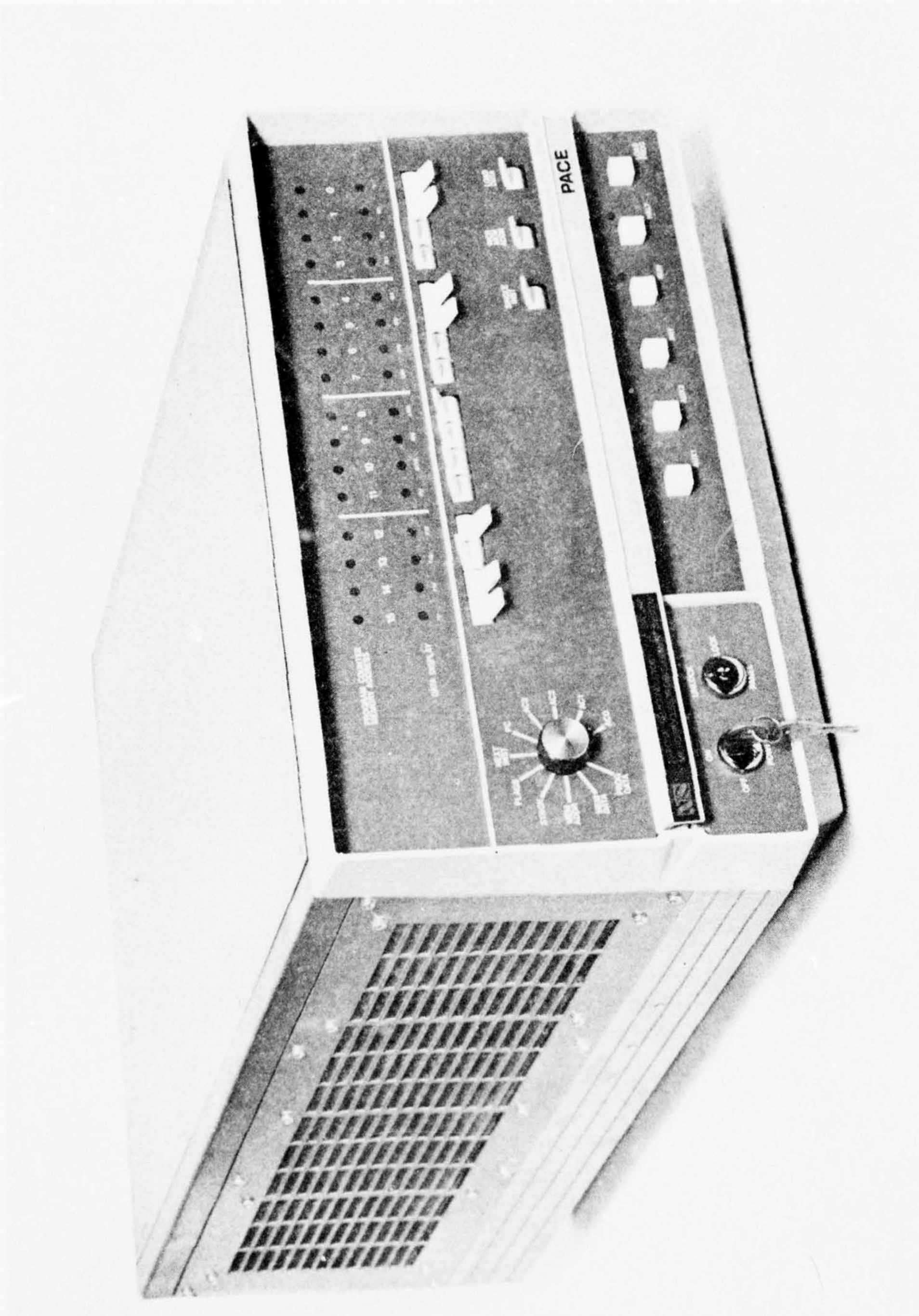

Figure 6. PACE MDS Showing Operator's Panel

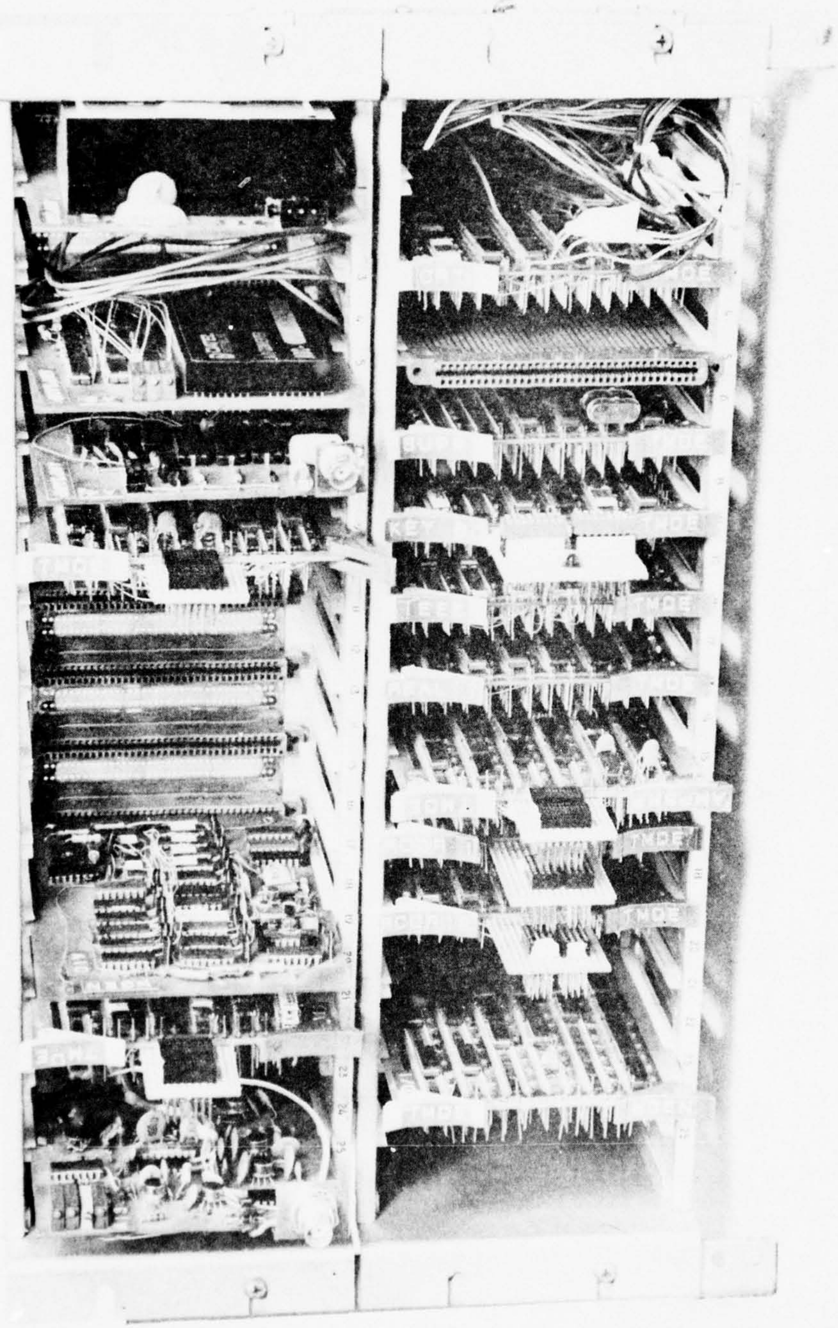

Figure 7. Double Card Cage Assembly

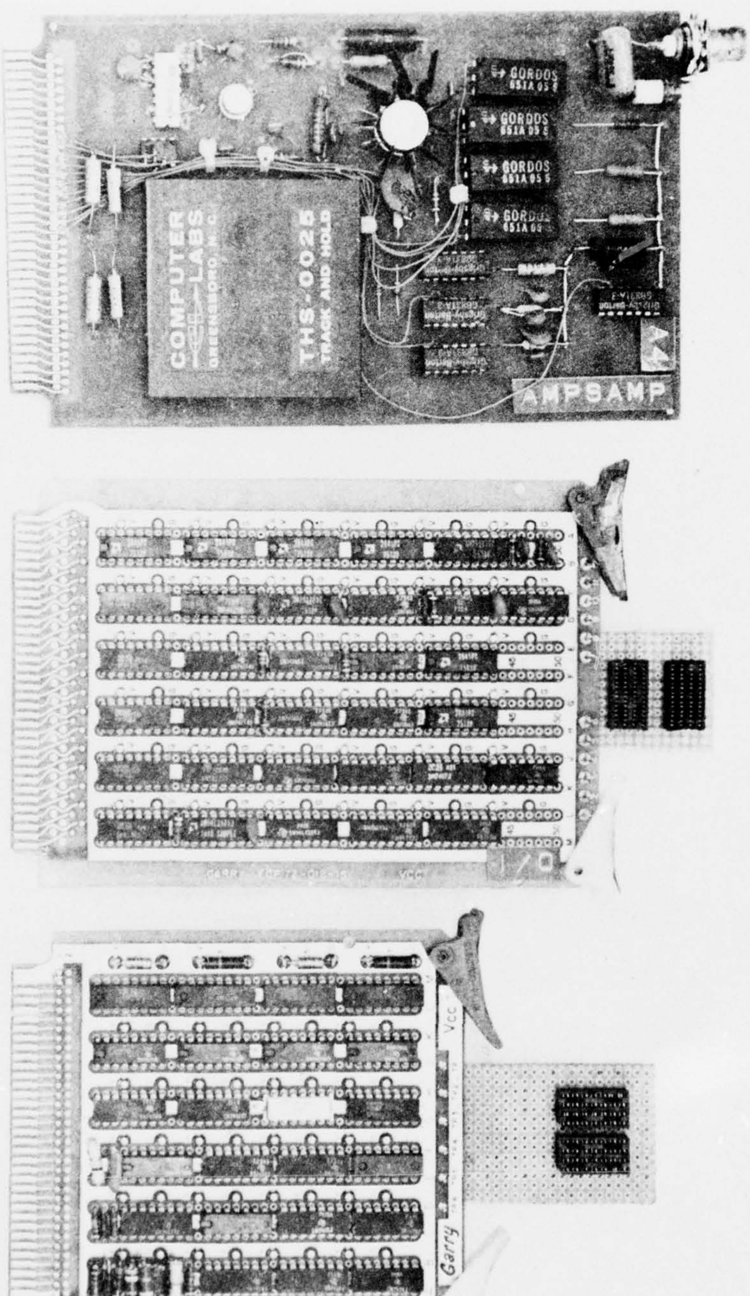

Figure 8. Examples of Cards Built for the Lab Model

the system. The middle card is one of the two more complicated, and therefore longer, wirewrapped digital interface cards in the system. The two larger interface cards in the system are used to interface the stimulus and sampling subsystems with the microprocessor. All the remaining interfaces were built on smaller wirewrap cards similar to the one on the right. The two sockets attached to the top of each interface card serve as I/O cable connectors. Each card has 72 contacts at the bottom. For interface cards, these pins are heavily used by a combination of microprocessor bus lines, additional lines from the CPU support board, and ground lines used for noise suppression.

It should be noted that the circuit construction techniques used on this project had both advantages and disadvantages. The advantages included ease and economy of construction and generous flexibility for making the frequent design changes required in an early stage of development. There were two main disadvantages. The first, and least immediate, was that the extensive use of wirewrapping caused the volume occupied by the circuitry to be much larger than it would have been using other techniques. This is not a problem for a lab model, of course, but it is one that will have to be addressed in the phototyping stage, if not earlier. The use of printed circuit, stitch welding, or other techniques can be used in the future to cut down volume requirements significantly. The second, and more immediate, disadvantage was the poor high frequency signal routing and noise suppression characteristics obtained. The performance obtainable from this packaging arrangement could certainly be improved upon with additional effort. Performance adequate for instrumentation applications will probably have to wait the arrival of brassboard or prototype construction techniques.

System Design

Figure 9 is a functional diagram of the TMDE lab model. It has been divided into three levels: one of hardware, one of software, and another of hardware. Several items of hardware, the microprocessor, its memory, and its bus, have not been explicity shown in this diagram but are assumed to be the medium of execution for the software blocks illustrated. The small rectangles labled INTERFACE represent the interface cards that plug into the lower card cage of the assembly shown in figure 7 which contains the microprocessor bus. This diagram is not useful for determining the location of I/O cables. For instance, the arrows connecting the interface blocks to the area labeled SOFTWARE are actually indicative of the fact that these cards plug directly into the microprocessor bus. On the other hand, even though the interface blocks in the diagram directly adjoin the device block which they interface to

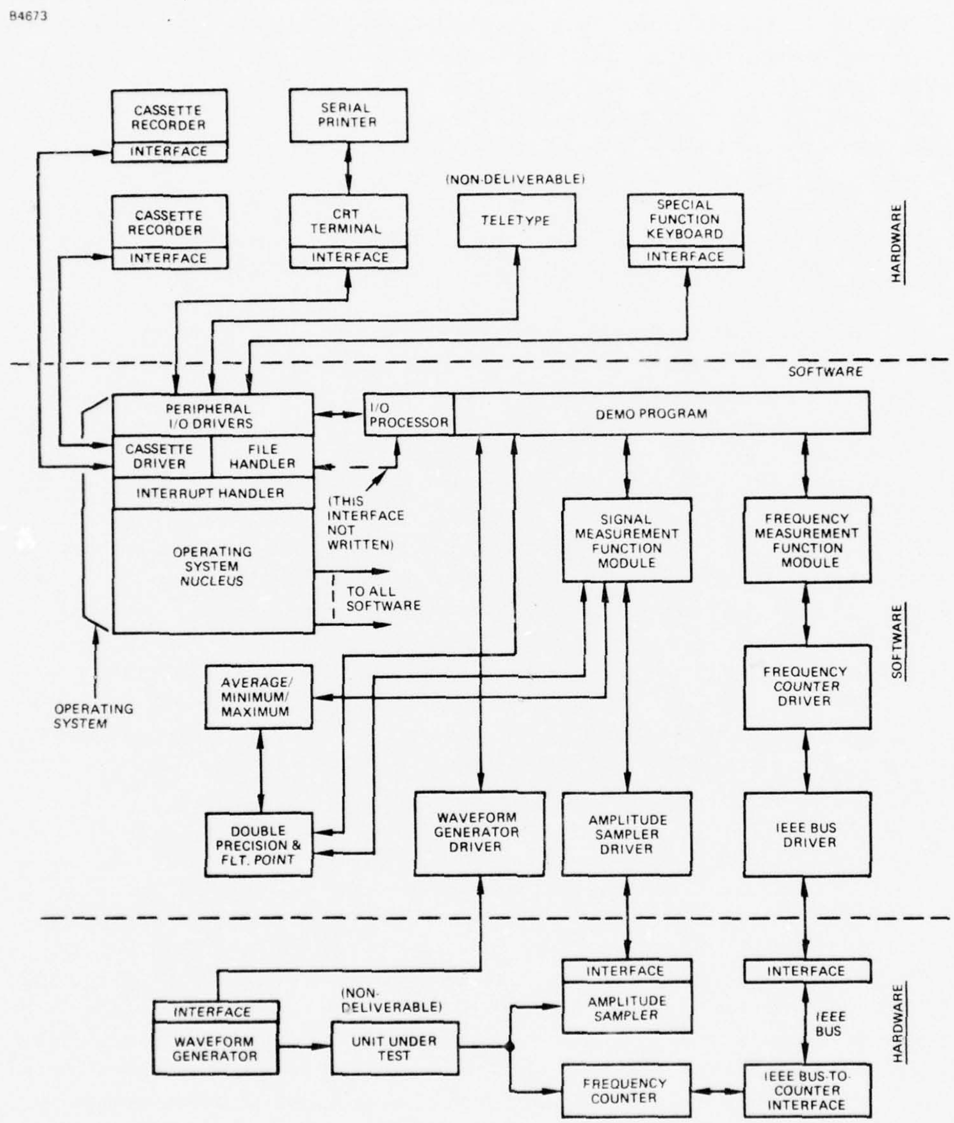

Figure 9. Functional Diagram of TMDE Lab Model

the microprocessor, they are physically connected in the lab model by I/O cables to their respective devices. The Teletype is not shown with an interface block because the PACE MDS comes equipped with its own Teletype interface.

The top level of the functional diagram contains the peripherals which allow the system to communicate with the operator. The bottom level contains the hardware which implements physical testing activities. The middle level contains the functional blocks of software whose functioning constitutes the bulk of the work performed by the total system and which gives the system most of its capabilities and characteristics as perceived by the human operator.

Hardware—Figure 2 is a block diagram of the TMDE lab model hardware. Functionally the hardware interfaces with the software as shown in figure 9. This section describes the function of the various hardware components and how they work together with each other and with the software. The hardware has been classified in three groups: the microprocessor, the hardware peripherals, and the stimulus and measurement hardware.

Microprocessor—The microprocessor in the lab model is required to support many types of functions—it must control and respond to various stimulus and measurement modules during test sequences, control any additional random logic required for system support, perform computational and data manipulation tasks, and interface with a human operator through periphrals. In order to keep the memory and software programming requirements for these areas to an acceptable minimum, emphasis during the selection process was placed on the power of the microprocessor's instruction set and the efficiency of its architecture in implementing the details of this system.

The National Semiconductor PACE microprocessor was selected as the device best fulfilling the above general criteria. PACE is a 16-bit microprocessor in a 40-pin ceramic package. It has four accumulators, a ten-word on-chip pushdown stack, six levels of vectored interrupt, DMA capability, an effective set of 45 instructions, and it addresses up to 64 K (K = 1, 024) 16-bit words. The high-yield PMOS technology used in fabricating this 16-bit microprocessor allows it to sell for less than competing 8-bit machines using NMOS technology.

Hardware Peripherals—Referring to the top level of figure 9; shown there are the two digital cassette tape recorders that were interfaced to the microprocessor. These recorders are general purpose; that is, they are

insensitive to data content. Thus, these recorders could be used to record the results of various equipment tests and diagnostics for later playback and display. In some applications of CARTE systems, a cassette could be used to store a related set of test programs. The operator could put this cassette into the transport and command the system to load the required programs into RAM memory one at a time for use as they are needed. In general, the cassette transports can be used to fulfill any system mass storage requirements for which fast access time is not a requirement. Cassettes were chosen as a possible mass storage medium because they are small and can be built to withstand shock and vibration.

The CRT terminal is used in this system as a substitute for what will probably be a smaller capacity, smaller volume flat panel display which will be used in a prototype model to provide an interactive mode of operation. Although a prototype or production model will have the small, built-in display for contact-type use, the interface for the CRT terminal can be retained so that the terminal can be plugged in when convenient to provide expanded output and, using its alphanumeric keyboard, enhanced input capabilities. The serial printer interfaced to the terminal provides hard copy capability under terminal keyboard control.

The ASR-33 Teletype was supplied by Melpar for the duration of the project. The PACE MDS has a built-in TTY interface and TTY I/O driver firmware in ROM's supplied with the unit. This made a TTY a good choice as an interactive I/O peripheral in the early stages of the project. The software development programs for the PACE MDS, such as the assembler and text editor, also use the TTY as a keyboard input device, a hard copy output device, and a mass storage device through the use of its paper tape punch and reader. A relay circuit had to be added to the TTY to enable the PACE MDS to start and stop the paper tape reader.

The special function keyboard is a device with two groups of 16 keys each, one group for initiating commands to the system with a single keystroke and one for entering numerical data into the system. A keyboard structure similar to this or some variation of it would probably be used in place of a full alphanumeric keyboard on a production model to both save panel space and to simplify operation. Both the data entry section and the command section of the keyboard were interfaced to the microprocessor bus, but no driver software was written for the data entry section since the TTY and the CRT terminal already provided a convenient means of implementing the same function. The command section interrupts the microprocessor at its highest user

interrupt priority. This gives the operator control over the system at all times, regardless of what function the system is engaged in. The command keys were used in the lab model to initialize the operating system, to select the I/O device currently in use, to control the modes of operation of the CRT display, and to start the execution of the demonstration program. In a production model these keys would be more directly involved in controlling the testing process.

Stimulus and Measurement Hardware—Since software replaces large amounts of hardware in the third generation approach, the stimulus and measurement subsystems should each be thought of as an inseparable combination of both hardware and software. That is, both subsystems are general purpose and programmable to the extent that their functions are determined almost wholly by the software that drives them. The following is a brief description of the techniques that implement this integrated hardware/software approach which reduces the quantity of hardware at the same time that it provides generalized capability.

The stimulus subsystem accepts from the computer a series of digital values which represent successive samples of the waveform to be generated. These samples are stored in the subsystem's own internal memory. Waveform parameters are also sent to the stimulus subsystem to determine such quantities as waveform frequency, amplitude, offset, and so forth. Once all data have been received, the stimulus subsystem adjusts its hardware to provide the required waveform parameters. It then sequentially outputs the digital values from its memory to a digital-to-analog converter, causing an analog approximation of the waveform to be constructed. The complete sequence of values is output repetitively at the rate necessary to cause the waveform frequency to be equal to the frequency requested by the computer. Through the use of this waveform synthesis technique, the computer can cause to be generated any periodic waveform that falls within the limits of the stimulus subsystem hardware.

The measurement subsystem hardware does basically one thing: It takes many analog samples of the signal being measured, converts the samples to digital numbers, and inputs these digital samples to the computer. No computation or effort at arriving at an actual measurement is made by the subsystem hardware. Once the computer has obtained the mass of raw data provided by the subsystem hardware, however, it can then apply its full computational power to this data through the use of numerical analysis techniques. Using this sample measurement approach, the computer can be programmed to make practically any desired measurements, subject to the limitations of the measurement subsystem hardware.

Software—Each block in the middle section of figure 9 represents a group of software which performs some well defined function. The blocks themselves fall into three different basic types of functions, that is Operating System, Demonstration Program, and Computational Software.

Operating System—The first type of function is performed by the operating system. Its main task is to provide a pool of software resources required in common by the programs that actually get the work done; that is, by the application software. (In this case the application software is all software except the operating system software.) The fact that these resources are available at all times makes it much easier for application software to function. In effect, the presence of an operating system creates a more sophisticated and powerful machine than that supplied by the bare hardware. As an example, there are no machine instructions which perform complete I/O functions with any of the hardware peripherals in the lab model. If all application programs were executed on the bare machine without benefit of an operating system, then each program that needed to communicate with the CRT terminal would have to have its own subroutine to perform I/O with the terminal. In fact, peripheral drivers have been written as part of the operating system which can be invoked by an application program almost as simply as executing a machine instruction. Further, the driver for the CRT terminal has been written to look like the driver for the TTY so that they are invoked in exactly the same manner. But instead of going directly to the TTY driver or the CRT driver program, an application program actually invokes an intermediate level program called an I/O "switch." The I/O switch program then passes the invocation on to either the TTY driver or to the CRT driver, depending on which one it was switched to at the time it was invoked by the application program. In this manner, application programs only have to make calls to one type of I/O device in order to communicate with the system operator, regardless of what device is actually being used. In the lab model, the operator can set the I/O switch to either device by pushing keys on the command section of the special function keyboard. This causes the special function keyboard driver program to determine which key was depressed and, if it was a key for I/O device selection, to set the I/O switch to the proper device.

The operating system software which handles the digital cassette tape transports is somewhat more involved than the other peripheral driver programs. First, a tape transport is a complicated device which requires much more detailed control than the other I/O peripherals. And second, it is a mass storage device on which many individual groups of information, known as files,

are written and selected in a random fashion for reading back to the microprocessor. Thus, the software for performing these functions is divided into two distinct parts. One, called the cassette driver, handles the details of actually controlling the transport, its speed, direction, whether it is reading or writing data, error checking, and so forth. The other part, called the file handler, is concerned with higher level functions. It does such things as breaking up data in a file into blocks of manageable size, adding header data to these blocks so that the data can be recognized on playback, recording data in the proper area on tape, and assigning unique file numbers for later file identification and recall purposes. On playback the file handler must search the tape for the requested file number, read it back, strip off header data from the data blocks, and present the retrieved data to the application program undisturbed in content or format. Both the cassette driver and file handler were written and tested. There was not enough memory space in the 4K (1K = 1,024 words) of memory allocated to the operating system to hold the file handler program. The program written to demonstrate the lab model's capabilities was not provided with an interface to the file handler.

Above and beyond making specific I/O capabilities available, the operating system also provides a multitasking environment for application software. Conceptually, this means that the software can be divided into parts, each with a particular task to perform, which can function as independent entities and can execute concurrently with each other. Since there is only one computer in the system, the term "concurrently" cannot mean simultaneously. It means that the parts of software, called "jobs" here, can take turns executing segments of their task on the one processor. Why should they break their tasks into segments? Why don't they complete their task in one continuous time frame? Well, they often do. But sometimes they execute to a certain point and then find they lack something which is required to complete their task. Perhaps they need to output data to a device which cannot presently accept any more data. Or maybe one job needs to receive intermediate results from another ongoing job. (Of course, the two jobs could have been written as one job so no discrete transfer of results is necessary, but this is often very undesirable from several standpoints, such as modularity, etc.). It is also possible that the job from which intermediate results are required is currently in use by yet another job, even though it is not presently executing. So when a job comes to a point in its task where it must wait on something, the multitasking environment provides the ability to stop execution of the present job and start another job which is ready to run. When the item which the job needs to continue its task becomes available, the job will then be allowed to run again. By multiplexing the processor among the jobs in this manner, multiple tasks can be in process concurrently. Thus, the microprocessor and other system

resources are utilized more efficiently by keeping them busy a larger percentage of the time than would otherwise be possible without a multitasking environment.

The part of the operating system which provides the multitasking environment is identified as the nucleus in figure 9. A connection is shown from the operating system nucleus to all other software because all application programs have been written as jobs which are designed specifically to run under the operating system nucleus. Several features were incorporated into the nucleus in order to further exploit the benefits of a multitasking environment. For instance, when one job stops executing because it must wait for some resource or because it has completed its task, the nucleus must then select a job to run next from among those jobs which are ready to run. This process, called job scheduling, is performed in a manner which allows jobs to be selected according to pre-assigned job priorities. This aids in regulating the competition for processor time and other system resources so that the more urgent tasks effectively receive higher throughput rates and quicker response times from the system than those tasks of lesser importance. Also, the software structure which responds to hardware interrupts has been designed to be compatible with and an integral part of the nucleus. When an interrupt occurs, the nucleus inputs a status word from the interrupting device and then makes the appropriate device service job ready to run. The service job competes for processor time on a priority basis along with all the other jobs in the system which are ready to run. When the device service job starts execution, it receives the device status word from the nucleus and then proceeds to service the device. It is possible that for some devices the status word can contain the actual input data word and thus no additional transactions between the device and its service job will be required. There may be devices used with this operating system in the future which require an interrupt service response time which is quicker than can be obtained by working through the operating system. If this were the case, the operating system could be bypassed and it would then be possible to start execution of a device service routine a maximum of 20 microseconds after the interrupt occurs. The possible side effects such a technique would have on the integrity of the nucleus have not been studied as yet, however. One final and very important feature which has been incorporated into the nucleus is the fact that jobs have been given the ability to communicate with each other through convenient, standardized means. This is what makes possible the division of a software system into manageable modules which cooperate with each other in the performance of the system's functions.

In summary, this project has resulted in a microprocessor operating system being written that should fulfill both the needs of the present and those of the foreseeable future. It provides a standard structure into which new software modules can easily be integrated to insure future expandability, and it provides a flexible, efficient environment for the operation of these modules. The operating system is described in more detail in the software documentation.

Demonstration Program—The second type of basic function performed by the software is that of interfacing all of the capability generated by the third generation type computational software to the system user in a convenient, easily used manner. This function is provided in the lab model software by the demonstration program and its I/O processor as shown in figure 9. It was thought to be important to keep the computational software separate from the operator interface. This was achieved by the simple measure of configuring the demonstration program as a self-contained job and interfacing it to the unified set of jobs that make up the computational software. Having done this, it is now possible to write different types of user interface software as desired and allow each to use the standard stimulus and measurement capabilities provided by the computational software and its associated hardware. For instance, it was originally proposed that the user interface to the system would take the form of an interpretive test language. The present form of the demonstration program was actually implemented, but if the interpreter were to be substituted for the demonstration program now, the only software that would be lost would be the demonstration program itself.

The demonstration program presents an interactive format to the system operator. This format consists of a dialogue between the operator and the system which begins with the operator entering two-letter mnemonic commands through the terminal keyboard and the demonstration program displaying, if necessary, a succession of one or more requests for additional information or parameters. After the operator enters the requested information in decimal numeric form, the demonstration program then executes the command. This may or may not then cause results to be displayed on the terminal, depending on the type of command entered.

The demonstration program contains a command line interpreter subprogram which inputs alphanumeric characters from the terminal via the I/O processor, determines which, if any, command has been entered, and then passes control to the subprogram responsible for carrying out all the functions associated with that command. This subprogram in turn calls on other subprograms in the demonstration program and on those jobs that make up the computational software which are needed in order to execute its particular command.

The I/O processor is one of the more complex of the subprograms that make up the demonstration program. It provides human-oriented interface capabilities for the program by performing various formatting and code conversion functions for both input and output. On input, the calling subprogram may request that an actual binary numeric value be obtained from the terminal. In this case, the I/O processor takes care of all the details involved in the process, including inputting individual characters from the terminal, performing various input data error checks, detecting end-of-input, and finally converting from ASCII-decimal to binary. On output, the calling subprogram may supply a number which is in either of two different internal formats and request that it be output in either decimal or hexadecimal form along with several parameters indicating output format and actions to be taken for various contingencies. These capabilities provide convenient system control and attractive, easily read system output.

Computational Software—The third type of basic function performed by the software shown in figure 9 is of a computational nature; that is, it is the part that actually does the work rather than provide system services or operator interface. Thus, the computational group of software is everything in the software area except for the operating system and the demonstration program. This software with its associated hardware is the implementation of the third generation ATE concepts referred to previously.

The general approach taken here to designing the software directly applicable to third generation techniques is to divide all the coding associated with a particular piece of hardware into two parts. One part, the hardware driver program, sets up and controls the hardware and transfers data to or from it as required. A second part, the function module, generates the data output to the hardware or analyzes the data input from the hardware by the driver program. In either case, it is the function modules that actually implement the required algorithms and initiate the number crunching required in order to finally provide the system's basic stimulus and measurement functions. In addition to these programs, there are a few others of a general support nature (such as the average minimum/maximum and double precision/floating point routines). The division between these programs was solidified by configuring each of them as a separate job to run under the operating system.

Waveform Generator Routine—The stated philosophy governing the partitioning of software was followed with varying degrees of success. The software dedicated to waveform generation was merged into a single job, called simply the waveform generator driver, instead of dividing it into one job for

waveform data generation and another for waveform generator hardware control. As a result, a call to this one job results in the generation of the desired waveform with its specified parameters by the waveform generator hardware.

Signal Measurement Routines—The partitioning of software for signal measurement functions went according to plan. A call to the signal measurement function module (SMFM) causes a set of parameters to be passed to the amplitude sampler driver. It in turn sets up the amplitude sampler with the proper parameters, such as input impedance, sampling rate, etc., and then starts the hardware operating. After the raw data has been collected, the driver job returns control again to the SMFM job. It then proceeds to process the raw data in order to produce final signal measurements, which are passed back to the demonstration program. The demonstration program may then output the resulting numerical values to the terminal with appropriate annotating comment and labels, or it may use these values in calculations of its own before anything is output. The latter occurs in the lab model when the system is commanded to run gain tests on an audio amplifier circuit.

Frequency Measurement Routines—Because an extra step was included in interfacing the frequency counter to the microprocessor, the software associated with frequency measurement ended up as three separate jobs. This extra interfacing step was the inclusion of an IEEE bus, Standard 488-1975. The bus was provided in order to easily take advantage of IEEE bus-compatible peripherals or devices in the future, if such inclusion were deemed desirable at that time. In this hierachy of software, the frequency measurement function module (FMFM) calls the frequency counter driver, which sets up the frequency counter to take the measurement. In order to do this, however, the frequency counter driver must send its commands over the IEEE bus. Thus, once it has its command words formatted for the frequency counter, it sends these command words to the IEEE bus driver. The IEEE bus driver in turn communicates over the IEEE bus interface which finally controls the frequency counter. Once the measurement is taken, the data is routed back over the same path to the FMFM where it is code-converted and checked for validity before being finally sent on to the demonstration program. Any drivers for future IEEE bus-compatible devices should be able to use the IEEE bus driver equally well.

SYSTEM PERFORMANCE

The TMDE lab model worked well. The hardware and software subsystems were successfully integrated into a smoothly functioning microprocessor-based test system possessing a basic set of stimulus, measurement, and testing capabilities. Due to limited funding and priority on software integration some individual operating discrepancies were not resolved, as described in the following paragraphs. Such discrepancies did not, however, detract from achieving the basic goals of the contract.

The waveform generator hardware produced the software-determined waveforms over a frequency range of a million to one with good amplitude and frequency resolution. At higher frequency ranges the waveform frequency actually produced is less than that requested by a fixed percentage and at the very highest frequencies the FIFO memories, which hold and circulate the digital waveform data for D/A conversion, tend to drop bits at these extreme data shifting rates. These discrepancies can easily be corrected with additional effort including minor modifications and improved components.

The amplitude sampler hardware functioned as planned under detailed supervision of the computer. Measurement parameters such as signal coupling, input impedance, voltage range and sampling rates to 500 kHz were among those variables put under software control. A nonlinear segment in the signal measurement process could be improved with additional work on the front-end circuitry ahead of the A/D converter. Also, in the same hardware, an excessive amount of high frequency digitally generated noise is currently present on the input signal to the A/D converter. Shielding and signal routing improvements would reduce this noise.

The operating system written for the microprocessor appeared to function with no problems. It performed its processor multiplexing and job communication functions flawlessly. All application software modules were integrated with each other through the operating system with little or no difficulty. There were some areas in the application software which generated a considerable amount of overhead execution time through the frequent, repetitious use of operating system functions. This was especially true regarding usage of the job which performed floating point arithmetic for all other software.

The application software worked well in providing the system with most of its functional capabilities. This is particularly significant for a system in

which the hardware is basically "dumb" and the software provides the required computational and decision-making capabilities. The demonstration program, which essentially interfaces the user with the rest of the system, provided a meaningful, easily understood method of accessing and controlling the details of system operation. Some delays were detectable between the issuance of a command and its completion, due to processing time requirements but none were considered to be bothersome.

Lab Model Capabilities

Table 1 contains a concise listing of the specific capabilities of the TMDE lab model's capabilities.

Performance Demonstration

Appendix A is a copy of the TMDE Lab Model Demonstration Procedure. It contains an explanation of the function and usage of each of the commands accepted by the system and then shows examples of output produced by the system in response to the commands. The system indicates its readiness to accept a command by outputting an asterisk on the CRT terminal screen.

The operator can then enter a command by typing in a two-letter mnemonic code on the terminal keyboard. Hard copy output of the contents of the terminal screen is made by a serial printer which interfaces directly to the CRT terminal. The system often asks the operator to pick one of several possible numbered parameters by typing the number of the desired parameter. If the operator hits the RETURN key without typing any number, this will be interpreted by the system as selecting the parameter numbered "zero." This is why some of the examples in the demonstration procedure appear to contain questions from the system that are unanswered by the operator: they were answered with the non-printing RETURN key. Additionally, the dc voltage notations beside the RD (raw data) command examples list the voltages measured from a laboratory voltage standard.

TABLE 1

TMDE LAB MODEL CAPABILITIES

Stimulus Capabilities

1. Operator Control: Via 2-letter mnemonic commands entered through CRT terminal keyboard.
2. Waveforms Available: Sine, square, ramp, and dc level
3. Frequency Range: 2 Hz to 2 MHz
4. Frequency Resolution: .01% (worst case)
5. Maximum Waveform Amplitude: 20 volts, peak-to-peak
6. Waveform Amplitude Resolution: 0.1% of quantization level
7. Waveform Quantization: 8 bits (256 levels)
8. Maximum DC Offset: ±10 volts
9. DC Offset Resolution: 0.01%

Measurement Capabilities

1. Operator Control: Via 2-letter mnemonic commands entered through CRT terminal keyboard
2. Measurements Available: Unprocessed samples of signal, average dc value, average ac value, ac maximum and minimum, peak-to-peak amplitude, frequency
3. Input Impedances: 50, 600, 100 K, and 1 M ohms
4. Analog-to-Digital Converter Resolution: 12 bits, or 0.024% of full range (3-1/2 digits)
5. Voltage Ranges: 0.25 volts, 2.5 volts, 25 volts, and 250 volts
6. Maximum Sample and Conversion Rate: 500,000/second
7. Maximum Number Consecutive Samples at 500 kHz: 127

Testing Capabilities

Presently programmed to automatically perform gain tests on an audio amplifier. Operator can specify a frequency range over which the test is made, and the frequency intervals in that range at which individual gain measurements are taken. The operator sets minimum and maximum allowable gains, and then asks for an output format that prints either all frequencies and gains, only those which are out of tolerance, or a simple pass/fail indication.

CONCLUSIONS

The basic conclusions drawn from the results of the work performed on this project are as follows:

a. Signal generation with fully programmable waveshape and frequency (up to 2 MHz) were achieved; frequencies up to about 10 MHz are practical using the current techniques.

b. The basic amplitude sampler hardware design is adequate but production units will require improved construction techniques to provide sufficient signal isolation and noise suppression.

c. The frequency counter interfaced to the system by the IEEE-488 bus worked. This arrangement is more of a second generation ATE configuration and was used to expedite the development of the lab model in place of taking the time to integrate the frequency counter function into the amplitude sampler. It also illustrated the potential of fourth generation architecture; that is, of distributed processing.

d. The lab model stimulus measurement hardware worked and demonstrated the feasibility of miniaturized ATE under microprocessor control.

e. The operating system written for the PACE microprocessor worked reliably. It provided a standardized environment for all application software modules, thereby encouraging program modularity and reducing interfacing difficulties. Its multitasking configuration, when used in conjunction with hardware interrupts, maximizes CPU utilization. Under some usage patterns, system overhead becomes relatively large.

f. The application software written for the PACE microprocessor worked well. By providing a moderate amount of both general purpose and specialized test capability in an interactive format, it further validated the concept of a small, microprocessor-based ATE system.

g. The software written for the lab model is contained in 12 K of 16-bit words of semiconductor RAM memory. This demonstrated that there is no particular problem in using relatively large amounts of memory with a microprocessor for a complex application, especially when the microprocessor is a 16-bit machine with minicomputer architecture, as is PACE.

In summary, this project demonstrated that the implementation of a physically small, automated TMDE system capable of being quickly adapted to perform analog as well as digital tests on different equipments in the field is highly practical and can be achieved using high density semiconductor memories and high performance microprocessors.

RECOMMENDATIONS

The results from this limited development effort indicate that the objectives and techniques investigated have great potential for solving many of the Army's current and future maintenance problems. The next step in exploiting this potential into an operational capability is to package the technology developed on this project into an integrated, portable brassboard model that can be demonstrated to potential users. Positive user acceptance would justify the funding of a pre-production prototype system.

The program to build a brassboard model of a CARTE (Contact and Repair Test Equipment) system based on microprocessor technology and using third generation ATE concepts can be accomplished by building on the groundwork provided by this project. The microprocessor feasibility has been demonstrated; follow-on effort should concentrate on upgrading the system's overall technical quality and capability to ensure a fair assessment by potential users. Accomplishing this will require significantly greater amounts of funding for additional work on both software and hardware than were available for this initial exploratory program.

Several areas of hardware will have to be further developed. Microprocessor and memory cards will have to be designed to take the place of the microprocessor development system currently used to perform these functions. An optimum density circuit card construction technique will have to be selected. Evaluations must be made of the system from a thermal engineering standpoint. The problem of high frequency noise suppression in the analog sampling circuits will require resolution in order to obtain good measurement accuracy from the system. The use of improved packaging techniques should minimize the seriousness of noise problems.

The optical isolators used in the amplitude sampler need to be replaced with currently available devices that can operate up to 10 MHz. A high speed (10 MHz) A/D converter can be used with the existing sample-and-hold circuit in the amplitude sampler to reduce the droop error caused by using the current 500 kHz A/D converter. The shift registers in the waveform generator should be replaced with ECL RAM's to eliminate the current propagation errors that are peculiar to the synthesizer shift register design. All circuits in the synthesizer section should be converted to ECL devices to eliminate the current problem of slow speed logic level converters to and from the TTL circuits. The waveform generator output analog offset and attenuator circuits should be modified to expand their dynamic range of operation. The frequency counter

and its remote interface to the IEEE bus should be replaced with a counter containing its own internal IEEE STD-488 interface.

The software now used in the TMDE lab model contains inefficient and limited capabilities due to limited funds. The following lists specific examples that should be improved upon:

a. The double precision integer arithmetic module as currently written is inefficient in terms of execution time. Parts of it are duplicated several places elsewhere in the software. These duplications should be eliminated and the module code tightened up.

b. The CRT driver program and the I/O switch program need to be rewritten to permit faster I/O transfers to take place.

c. The floating point arithmetic module is currently configured as a system job. It should be rewritten as a directly callable re-entrant subroutine in order to avoid the overhead generated by frequent, repetitious system calls.

d. The cassette driver and file handler need additional work to make them more flexible and general purpose. Interfaces to the file handler should be added to the I/O processor module, which is part of the demonstration program.

e. The I/O processor module should be removed from the demonstration program and made part of the operating system so its resources will be available to all future software.

f. Totally new software modules will have to be written for the system. A square root subroutine should be added in order to provide true RMS measurement capability. A Fast Fourier Transform (FFT) capability will be needed to provide frequency-related measurements. These and other additions to the software will be required to make it compatible with the levels of performance intended for the brassboard model.

The addition of an FFT capability to the TMDE lab model will greatly enlarge the scope of the lab model's measurement capabilities. It will provide a means to perform measurements of harmonic distortion and S/N ratios, and to implement selective RF voltmeter capabilities. Measurements of this type are limited to signals under 250 kHz using the present A/D converter. Faster converters exist, but they rapidly increase in size and/or cost as their

conversion rate increases. In order to provide measurement capabilities of the sample measure type up to 100 MHz, newer technologies which are now emerging will have to be applied to this part of the system.

The lab model currently has the potential capability to generate any type of waveform at frequencies into the low megahertz region. Signal generation above this frequency range is usually limited in application to communication system testing and requires only a few basic waveforms, such as sine waves and square waves. It is possible to produce these waveforms into the 100 MHz region with good frequency resolution and fast frequency slewing by going to direct generation techniques instead of the waveforms synthesis technique currently used in the TMDE lab model. It is also practical to provide both AM and FM modulation in this frequency range. Accomplishing these goals will require some specialized hardware apart from the general purpose, waveform synthesis generation hardware. It should be practical, however, to make this hardware miniaturized and computer controlled so that it fits into the current philosophy of CARTE systems.

The hardware and software should both be developed into a near final form as they might be used in a production unit. The brassboard model should incorporate the concept of providing a basic hardware mainframe into which is plugged the minimum set of standard modular hardware resources required for a given maintenance task. Figure 10 is an artist's concept of one possible implementation of a CARTE system intended for general applications. Different types and sizes of mainframes could be utilized and yet retain compatibility by virtue of the fact that they all accept the same standard, modular hardware stimulus and measurement components. This would be made possible by the use of a standard bus in all mainframes into which the modules are plugged. A bus and packaging standard currently exists which exhibits the characteristics needed by a flexible, high performance instrumentation system operating under computer control. This is the CAMAC (Computer Automated Measurement and Control) System, which was developed in the process control and nuclear instrumentation industries and has been adopted as a standard (583) by the IEEE. At the present time the CAMAC mechnical packaging standard appears to be suitable for a brassboard model for cost saving. The use of the CAMAC electrical bus is dependent on a tradeoff between what would be saved by using commercially available CAMAC modules (now of second generation ATE type) along with modifying sections of the lab model software to use these modules versus adapting the current electrical designs as CAMAC modules for CARTE and continuing to use the microprocessor I/O bus. At the present time the latter plan appears to be a better way of going, using only the CAMAC physical design for packaging.

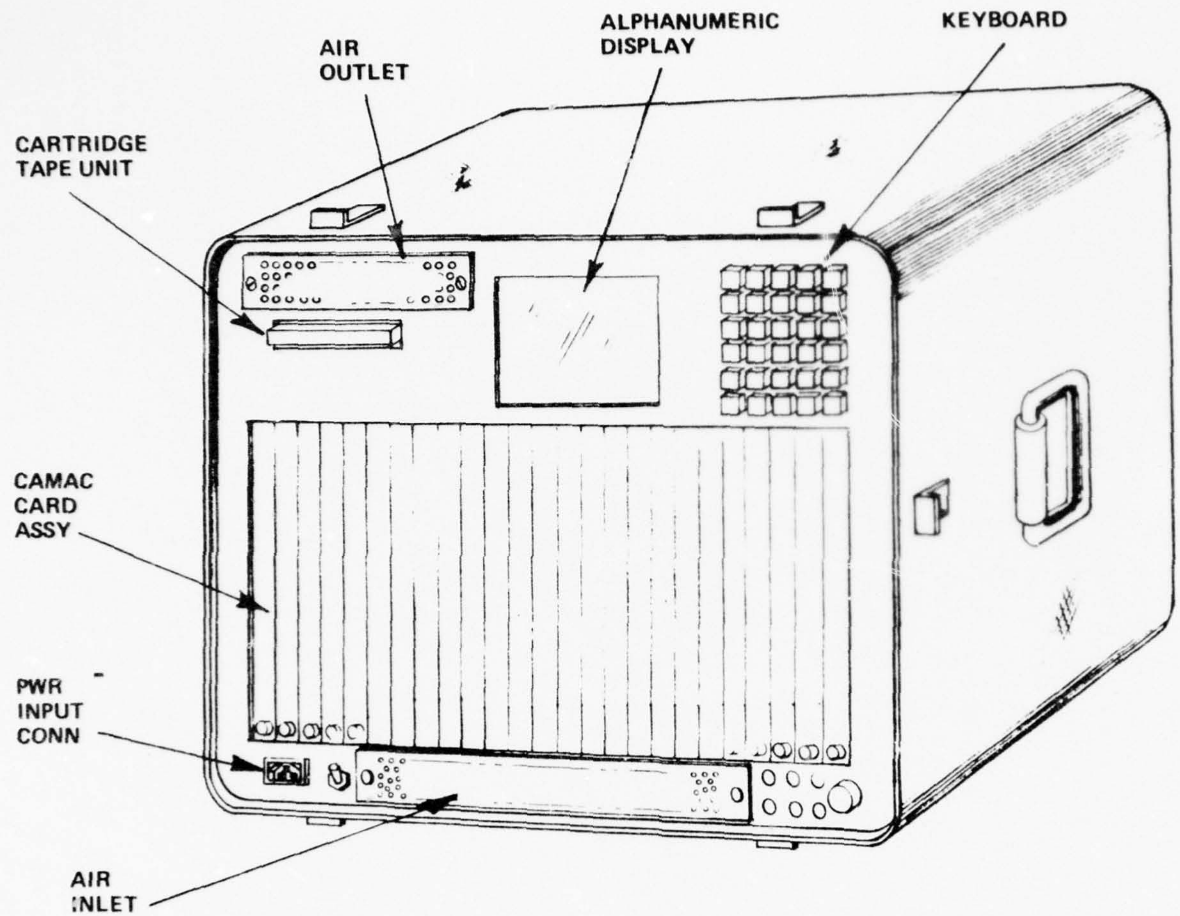

Figure 10. A CARTE System for General Applications

APPENDIX A

TMDE LAB MODEL DEMONSTRATION PROCEDURE

This procedure is designed to demonstrate capabilities available on the current version of the TMDE lab model. These capabilities are made available through two-letter mnemonic commands typed in at the terminal. The procedure will be oriented around these commands. The following procedure is to be followed after all equipment has been powered up and initialized and all programs have been loaded.

(1) ST: Permits the operator to set up one of four stimulus waveforms and their associated parameters.

(2) ON: Causes the waveform generator to actually generate the waveform set up by the previous ST command.

(3) OF: Turns off the waveform generator.

<u>Procedure</u>: Connect the output of the waveform generator to a scope. Type ST and RETURN. Select any waveform except DC LEVEL and all parameters asked for. Now type ON and RETURN, and observe that the waveform appears. Type OF and RETURN and observe that the waveform disappears. Type ON and then select other waveforms and parameters with the ST command.

```
*ST
WAVEFORM = (0)SINE, (1)SQUARE, (2)RAMP, OR (3)DC LEVEL? 0
FREQ (HZ) = 1000
AMPLITUDE (P-P MV) = 3000
DC OFFSET (MV) = 1000
OUTPUT OHMS = (0)50 OR (1)600? 0

*ON

*OF

*ON

*ST
WAVEFORM = (0)SINE, (1)SQUARE, (2)RAMP, OR (3)DC LEVEL? 1
FREQ (HZ) = 5000
AMPLITUDE (P-P MV) = 4000
DC OFFSET (MV) = -2000
OUTPUT OHMS = (0)50 OR (1)600? 1

*
```

(4) TR: Causes the system to use the frequency counter to perform a frequency measurement of the input signal every time it samples the signal so it can determine the correct sampling rate.

(5) FX: Allows the operator to input a frequency which the system will assume, for sampling rate purposes, to be the input signal frequency.

(6) FR: Causes the system to output to the terminal the frequency it is using for sampling rate determination. It will print (TR) or (FX) after the frequency to indicate the source of the frequency.

Procedure: Connect an AC signal source to the amplitude sampler, which should be daisy chained to the frequency counter. Type TR and RETURN. Then type FR and observe the output. Type FX and input some different frequency through the keyboard. Then type FR and observe the output.

```
*TR

*FR
FREQ =        5001 HZ (TR)
```

```
*FX
FREQ (HZ) = 5000

*FR
FREQ =        5000 HZ (FX)
```

(7) LD: Allows the operator to select the input impedance of the amplitude sampler so as to present various loads to the UUT. The default value after system initialization is 1 megohm.

Procedure: Type LD and select the desired load.

```
*LD
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 0

*LD
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 1

*LD
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 2

*LD
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 3
```

(8) RD: Causes the system to display raw data. That is, it outputs to the terminal the actual values of voltage for each sample it takes of the input signal.

Procedure: Input a known DC voltage to the amplitude sampler and type RD. Then select: (A) output to terminal, (B) fixed sample intervals, and (C) the desired number of samples (max number = 120). Observe the resulting output to the terminal. Now input a ramp to the amplitude sampler and repeat the procedure.

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
     1    19870 MV
     2    19990 MV
     3    20000 MV
     4    19710 MV
     5    19500 MV
     6    19480 MV
     7    19780 MV
     8    20030 MV
```

20 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1    10090 MV
    2     9800 MV
    3     9890 MV
    4     9830 MV
    5     9910 MV
    6    10040 MV
    7     9990 MV
    8     9720 MV

*
```

10 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1     5260 MV
    2     5050 MV
    3     5120 MV
    4     5600 MV
    5     5590 MV
    6     5290 MV
    7     5480 MV
    8     5660 MV

*
```

5 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      1872 MV
    2      1750 MV
    3      1829 MV
    4      1809 MV
    5      1833 MV
    6      1773 MV
    7      1824 MV
    8      1790 MV

*
```

2 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      858 MV
    2      843 MV
    3      916 MV
    4      843 MV
    5      903 MV
    6      909 MV
    7      905 MV
    8      845 MV

*
```

1.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      408 MV
    2      384 MV
    3      437 MV
    4      417 MV
    5      418 MV
    6      410 MV
    7      510 MV
    8      427 MV

*
```

0.5 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      203 MV
    2      205 MV
    3      203 MV
    4      205 MV
    5      204 MV
    6      206 MV
    7      202 MV
    8      205 MV

*
```

0.2 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
     1       103 MV
     2       105 MV
     3       103 MV
     4       105 MV
     5       102 MV
     6       103 MV
     7       104 MV
     8       104 MV

*
```

0.1 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
     1         3 MV
     2         2 MV
     3         4 MV
     4         5 MV
     5         6 MV
     6         4 MV
     7         3 MV
     8         1 MV

*
```

+0.0 VOLTS DC INPUT

```
RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =                                   -0.0 VOLTS DC INPUT
    1       -5 MV
    2       -4 MV
    3       -5 MV
    4       -4 MV
    5       -4 MV
    6       -4 MV
    7       -6 MV
    8       -5 MV

*
```

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =                                   -0.1 VOLTS DC INPUT
    1     -107 MV
    2     -106 MV
    3     -108 MV
    4     -107 MV
    5     -109 MV
    6     -110 MV
    7     -110 MV
    8     -107 MV

*
```

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      -207 MV
    2      -208 MV
    3      -207 MV
    4      -208 MV
    5      -207 MV
    6      -211 MV
    7      -209 MV
    8      -208 MV

*
```

-0.2 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1      -432 MV
    2      -499 MV
    3      -485 MV
    4      -423 MV
    5      -465 MV
    6      -452 MV
    7      -467 MV
    8      -412 MV

*
```

-0.5 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
   1      -943 MV
   2      -988 MV
   3      -982 MV
   4      -954 MV
   5      -939 MV
   6     -1016 MV
   7     -1009 MV
   8      -914 MV

*
```

-1.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
   1     -1872 MV
   2     -1844 MV
   3     -1846 MV
   4     -1926 MV
   5     -1877 MV
   6     -1896 MV
   7     -1882 MV
   8     -1851 MV

*
```

-2.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
     1     -5670 MV
     2     -5820 MV
     3     -5550 MV
     4     -5630 MV
     5     -5380 MV
     6     -5490 MV
     7     -5920 MV
     8     -5610 MV

*
```

-5.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
     1    -10330 MV
     2    -10090 MV
     3     -9880 MV
     4    -10110 MV
     5    -10020 MV
     6    -10270 MV
     7    -10290 MV
     8    -10090 MV

*
```

-10.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 8
FREQ =
    1   -19930 MV
    2   -20380 MV
    3   -20390 MV
    4   -20110 MV
    5   -19890 MV
    6   -19780 MV
    7   -20190 MV
    8   -20230 MV

*
```

-2.0 VOLTS DC INPUT

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 32
FREQ =          1000 HZ (FX)          RAMP, 5 VOLTS P TO P
     1       1549 MV                        NO OFFSET
     2       2077 MV
     3       1715 MV
     4       1146 MV
     5        611 MV
     6         47 MV
     7       -501 MV
     8      -1068 MV
     9      -1634 MV
    10      -2137 MV
    11      -1680 MV
    12      -1141 MV
    13       -570 MV
    14         -6 MV
    15        538 MV
    16       1104 MV
    17       1667 MV
    18

    17       1667 MV
    18       2066 MV
    19       1573 MV
    20       1048 MV
    21        479 MV
    22        -62 MV
    23       -626 MV
    24      -1202 MV
    25      -1717 MV
    26      -2117 MV
    27      -1629 MV
    28      -1070 MV
    29       -520 MV
    30         32 MV
    31        579 MV
    32       1119 MV
```

```
*RD
TO (0)TERMINAL OR (1)TAPE?
USE (0)FIXED OR (1)RANDOM SAMPLE INTERVALS?
NO. SAMPLES = 32
FREQ =          1000 HZ (FX)
    1       2126 MV
    2       1868 MV
    3       1297 MV
    4        560 MV
    5       -255 MV
    6      -1087 MV
    7      -1724 MV
    8      -2142 MV
    9      -2174 MV
   10      -1885 MV
   11      -1306 MV
   12       -495 MV
   13        324 MV
   14       1135 MV
   15       1765 MV
   16       2097 MV
   17       2105 MV
   18       1771 MV

   19       1184 MV
   20        388 MV
   21       -450 MV
   22      -1242 MV
   23      -1842 MV
   24      -2171 MV
   25      -2147 MV
   26      -1771 MV
   27      -1149 MV
   28       -344 MV
   29        447 MV
   30       1202 MV
   31       1794 MV
   32       2086 MV
```

SINEWAVE
5 VOLTS P-P
NO OFFSET

(9) DC: Samples the signal DC coupled and outputs the average DC value of the signal.

(10) AC: Samples the signal AC coupled and outputs the AC waveform's peak-to-peak value, its average amplitude, and its maximum and minimum values.

(11) ME: Makes a complete measurement. It takes all the measurements and outputs all the results of both "DC" and "AC."

Procedure: Set up the waveform generator to output a sine wave with some DC offset using the ST command. Route this signal to the input of the amplitude sampler. Type DC and observe the output. Type AC and observe the output. Type ME and observe the combined output types.

```
*ST
WAVEFORM = (0)SINE, (1)SQUARE, (2)RAMP, OR (3)DC LEVEL? 0
FREQ (HZ) = 1000
AMPLITUDE (P-P MV) = 5000
DC OFFSET (MV) = 1000
OUTPUT OHMS = (0)50 OR (1)600? 0

*DC
DC     =      750 MV
FREQ =       1000 HZ (FX)

*AC
P-P    =     4360 MV
AVG    =     1370 MV
AC MAX =     2159 MV
AC MIN =    -2200 MV
FREQ =       1000 HZ (FX)

*


*ME
DC     =      760 MV
P-P    =     4374 MV
AVG    =     1378 MV
AC MAX =     2170 MV
AC MIN =    -2204 MV
FREQ =       1000 HZ (FX)
```

(12) T1: This command enables an amplifier circuit to be tested for frequency response over a single continuous frequency band. It gives the user, once he has set up the test, the ability to quickly repeat the test. The operator also has the option, between each test, to alter the test parameters, the printout format, or both. Regardless of printout format, the operator always gets a pass/fail indication at the end of the test.

Procedure: Connect the waveform generator to the amplifier input, and the amplifier output to the amplitude sampler. Apply power to the amplifier. Stimulate the amplifier with a 50 ohm, 1000 Hz, 500 MV sine wave using the ST command and adjust the gain of the amplifier so its output is 5000 MV, giving an unloaded gain of 10. Now type T1 and supply all parameters. Specifically, the amplitude should be 500 MV, generator output impedance should be 50 ohms, loading should be 1 megohm, UUT input impedance should be 1 megohm, and its output impedance should be 10 ohms. Start the frequency sweep at 50 Hz, use a frequency increment of 1000 Hz, and use a stop frequency of 16000 Hz. Use appropriate minimum and maximum gains, such as 8 and 12, respectively, to provide a range of acceptable values around the nominal gain of 10. Ask for all results to be printed. Repeat the test with different parameters and output formats.

```
*T1
ENTER/CHANGE TEST SETUP (0)NO OR (1)YES = 1
ENTER NEW TEST PARAMETERS (0)NO OR (1)YES = 1
AMPLITUDE (P-P MV) = 500
DC OFFSET (MV) = 0
OUTPUT OHMS = (0)50 OR (1)600? 0
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 0
UUT INPUT IMPEDANCE (OHMS) = 1000000
UUT OUTPUT IMPEDANCE (OHMS) = 10
START FREQ (HZ) = 10
INCREMENT  (HZ) = 1000
STOP FREQ  (HZ) = 16000
MIN GAIN = 6
MAX GAIN = 10
PRINT ALL FREQ & GAINS? (0)NO OR (1)YES = 1
```

```
BEGIN TEST
     FREQ        GAIN
       10           1 FAILED LOW
     1010           8
     2010           8
     3010           8
     4010           8
     5010           7
     6010           7
     7010           6
     8010           6
     9010           6
    10010           5 FAILED LOW
    11010           5 FAILED LOW
    12010           5 FAILED LOW
    13010           4 FAILED LOW
    14010           4 FAILED LOW
    15010           4 FAILED LOW
UNIT FAILED
END TEST
```

```
RUN TEST AGAIN? (0)YES OR (1)NO
ENTER/CHANGE TEST SETUP (0)NO OR (1)YES = 1
ENTER NEW TEST PARAMETERS (0)NO OR (1)YES = 1
AMPLITUDE (P-P MV) = 500
DC OFFSET (MV) = 0
OUTPUT OHMS = (0)50 OR (1)600? 0
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 0
UUT INPUT IMPEDANCE (OHMS) = 1000000
UUT OUTPUT IMPEDANCE (OHMS) = 10
START FREQ (HZ) = 10
INCREMENT  (HZ) = 100
STOP FREQ  (HZ) = 1000
MIN GAIN = 6
MAX GAIN = 10
PRINT ALL FREQ & GAINS? (0)NO OR (1)YES = 1



BEGIN TEST
     FREQ        GAIN
       10           1 FAILED LOW
      110           8
      210           8
      310           8
      410           8
      510           8
      610           8
      710           8
      810           8
      910           8
UNIT FAILED
END TEST

RUN TEST AGAIN? (0)YES OR (1)NO
```

```
*T1
AMPLITUDE (P-P MV) = 500
DC OFFSET (MV) = 0
OUTPUT OHMS = (0)50 OR (1)600? 0
LOADING (OHMS) = (0) 1M, (1) 50, (2) 600, OR (3) 100K? 0
UUT INPUT IMPEDANCE (OHMS) = 1000000
UUT OUTPUT IMPEDANCE (OHMS) = 10
START FREQ (HZ) = 10
INCREMENT  (HZ) = 10
STOP FREQ  (HZ) = 110
MIN GAIN = 6
MAX GAIN = 10
PRINT ALL FREQ & GAINS? (0)NO OR (1)YES = 1

BEGIN TEST
     FREQ        GAIN
       10          1 FAILED LOW
       20          2 FAILED LOW
       30          4 FAILED LOW
       40          5 FAILED LOW

       50          6
       60          6
       70          7
       80          7
       90          7
      100          7
      110          7
UNIT FAILED
END TEST
```

(13) QU: This causes the demonstration program to quit execution.

Procedure: Type QU. Hit the EXEC key on the special function keyboard to resume execution.

DISTRIBUTION LIST

101 Defense Documentation Center
Attn: DDC-TCA
Cameron Station (Bldg 5)
012 Alexandria, VA 22314

103 Code R123, Tech Library
DCA Defense Comm Engrg Ctr
1860 Wiehle Ave
001 Reston, VA 22090

104 Defense Communications
Agency
Technical Library Center
Code 205 (P. A. TOLOVI)
001 Washington, DC 20305

200 Office of Naval Research
Code 427
001 Arlington, VA 22217

203 GIDEP Engineering & Support
Dept
TE Section
PO Box 398
001 Norco, CA 91760

205 Director
Naval Research Laboratory
Attn: Code 2627
001 Washington, DC 20375

206 Commander
Naval Electronics Laboratory
Center
Attn: Library
001 San Diego, CA 92152

207 CDR, Naval Surface Weapons
Center
White Oak Laboratory
Attn: Library, Code WX-21
001 Silver Spring, MD 20910

210 Commandant, Marine Corps
Hq, US Marine Corps
Attn: Code LMC
002 Washington, DC 20380

211 Hq, US Marine Corps
Attn: Code INTS
001 Washington, DC 20380

212 Command, Control & Com-
munications Div
Development Center
Marine Corps Development &
Educ Comd
001 Quantico, VA 22134

301 Rome Air Development Center
Attn: Documents Library (Tild)
001 Griffiss AFB, NY 13441

307 HQ ESD (DRI)
L. G. Hanscom AFB
001 Bedford, MA 01731

314 HQ, Air Force Systems
Command
Attn: DLCA
Andrews AFB
001 Washington, DC 20331

DISTRIBUTION LIST (Continued)

403 CDR, US Army Missile Command
Redstone Scientific Info Center
Attn: Chief, Document Section
002 Redstone Arsenal, AL 35809

404 Commander
US Army Missile Command
Attn: DRSMI-RE (Mr. Pittman)
001 Redstone Arsenal, AL 35809

405 Commander
US Army Aeromedical Research Lab
Attn: Library
001 Fort Rucker, AL 36362

406 Commandant
US Army Aviation Center
Attn: ATZQ-D-MA
003 Fort Rucker, AL 36362

407 Director, Ballistic Missile Defense
Advanced Technology Center
Attn: ATC-R, PO Box 1500
001 Huntsville, AL 35807

408 Commandant
US Army Military Police School
Attn: ATSJ-CD-M-C
003 Fort McClellan, AL 36201

417 Commander
US Army Intelligence Center & School
Attn: ATSI-CD-MD
002 Fort Huachuca, AZ 85613

418 Commander
HQ Fort Huachuca
Attn: Technical Reference Div
001 Fort Huachuca, AZ 85613

419 Commander
US Army Electronic Proving Ground
Attn: STEEP-MT
002 Fort Huachuca, AZ 85613

420 Commander
USASA Test & Evaluation Center
Attn: IAO-CDR-T
001 Fort Huachuca, AZ 85613

421 Commander
HQ US Army Communications Command
Attn: CC-OPS-SM
001 Fort Huachuca, AZ 85613

437 Deputy for Science & Technology Office, Assist Sec Army (R&D)
001 Washington, DC 20310

438 HQDA (DAMA-ARP/DR. F. D. Verderame)
001 Washington, DC 20310

455 Commandant
US Army Signal School
Attn: ATSN-CTD-MS
001 Fort Gordon, GA 30905

DISTRIBUTION LIST (Continued)

470 Director of Combat Developments
US Army Armor Center
Attn: ATZK-CD-MS
002 Fort Knox, KY 40121

473 Commandant
US Army Ordnance School
Attn: ATSL-CD-DR
002 Aberdeen Proving Ground, MD 21005

475 CDR, Harry Diamond Laboratories
Attn: Library
2800 Powder Mill Road
001 Adelphi, MD 20783

477 Director
US Army Ballistic Research Labs
Attn: DRXBR-LB
001 Aberdeen Proving Ground, MD 21005

481 Harry Diamond Laboratories, Dept of Army
Attn: DRXDO-RCB (Dr. J. Nemarich)
2800 Powder Mill Road
001 Adelphi, MD 20783

482 Director
US Army Materiel Systems Analysis Acty
Attn: DRXSY-T
001 Aberdeen Proving Ground, MD 21005

499 Commander
US Army Tank-Automotive Command
Attn: DRDTA-RH
001 Warren, MI 48090

507 CDR, US Army Aviation Systems Command
Attn: DRSAV-G
PO Box 209
001 St. Louis, MO 63166

511 Commander, Picatinny Arsenal
Attn: SARPA-FR-S
Bldg 350
002 Dover, NJ 07801

512 Commander
Picatinny Arsenal
Attn: SARPA-ND-A-4 (Bldg 95)
001 Dover, NJ 07801

513 Commander
Picatinny Arsenal
Attn: SARPA-TS-S #59
001 Dover, NJ 07801

515 Project Manager, REMBASS
Attn: DRCPM-RBS
002 Fort Monmouth, NJ 07703

517 Commander
US Army Satellite Communications Agcy
Attn: DRCPM-SC-3
002 Fort Monmouth, NJ 07703

DISTRIBUTION LIST (Continued)

518 TRI-TAC Office
Attn: CSS (Dr. Pritchard)
001 Fort Monmouth, NJ 07703

531 CDR, US Army Research Office
Attn: DRXRO-IP
PO Box 12211
001 Research Triangle Park, NC 07709

532 CDR, US Army Research Office
Attn: DRXRO-PH (Dr. R. J. Lontz)
PO Box 12211
001 Research Triangle Park, NC 27709

533 Commandant
US Army Inst for Military Assistance
Attn: ATSU-CTD-MO
001 Fort Bragg, NC 28307

537 CDR, US Army Tropic Test Center
Attn: STETC-MO-A (Tech Library)
Drawer 942
001 Fort Clayton, Canal Zone 09827

543 Division Chief
Meteorology Division
Counterfire Department
002 Fort Sill, OK 73503

554 Commandant
US Army Air Defense School
Attn: ATSA-CD-MC
001 Fort Bliss, TX 79916

555 Commander
US Army Nuclear Agency
001 Fort Bliss, TX 79916

556 Commander, HQ Masster
Technical Information Center
Attn: Mrs. Ruth Reynolds
001 Fort Hood, TX 76544

563 Commander, DARCOM
Attn: DRCDE
5001 Eisenhower Ave
001 Alexandria, VA 22333

564 CDR, US Army Security Agency
Attn: IARDA-IT
Arlington Hall Station
001 Arlington, VA 22212

568 Commander
US Army Mobility Eqp Res & Dev Cmd
Attn: DRXFB-R
001 Fort Belvoir, VA 22060

569 Commander
US Army Engineer Topographic Labs
Attn: ETL-TD-EA
001 Fort Belvoir, VA 22060

DISTRIBUTION LIST (Continued)

572 Commander
US Army Logistics Center
Attn: ATCL-MC
002 Fort Lee, VA 22801

602 Director, Night Vision Laboratory
US Army Electronics Command
Attn: DRSEL-NV-D
001 Fort Belvoir, VA 22060

603 CDR/DIR, Atmospheric Sciences Laboratory
US Army Electronics Command
Attn: DRSEL-BL-SY-S
001 White Sands Missile Range, NM 88002

604 Chief
Ofc of Missile Electronic Warfare
Electronic Warfare Lab, ECOM
001 White Sands Missile Range, NM 88002

606 Chief
Intel Materiel Dev & Support Ofc
Electronic Warfare Lab, ECOM
001 Fort Meade, MD 20755

680 Commander
US Army Electronics Command
000 Fort Monmouth, NJ 07703

1 DRSEL-PL-ST
1 DRSEL-NL-D
1 DRSEL-WL-D
1 DRSEL-VL-D
3 DRSEL-CT-D
1 DRSEL-BL-D
1 DRSEL-SA-L
1 DRSEL-CE
1 DRSEL-MA-D
1 DRSEL-TL-DT
3 DRSEL-TL-M
15 DRSEL-TL-MI (Ofc of Record)
1 DRSEL-TE
1 DRSEL-MA-MP
2 DRSEL-MS-TI
1 DRSEL-GG-TD
1 DRSEL-PP-I-PI
1 DRSEL-GS-H
1 DRSEL-CG (Mr. Doxey)
2 DRSEL-PA
1 DRSEL-RD
1 DRSEL-TL-D
1 USMC-LNO
1 TRADOC-LNO

700 CINDAS
Purdue Industrial Research Park
2595 Yeager Road
001 W. Lafayette, IN 47096

701 MIT - Lincoln Laboratory
Attn: Library (Rm A-082)
PO Box 73
002 Lexington, MA 02173

703 NASA Scientific & Tech Info Facility
Baltimore/Washington Intl Airport
001 PO Box 8757, MD 21240

DISTRIBUTION LIST (Continued)

704 National Bureau of Standards
Bldg 225, Rm A-331
Attn: Mr. Leedy
001 Washington, DC 20231

705 Advisory Group on Electron Devices
201 Varick Street, 9th Floor
002 New York, NY 10014

706 Advisory Group on Electron Devices
Attn: Secy, Working Group D (Lasers)
201 Varick Street
002 New York, NY 10014

707 TACTEC
Battelle Memorial Institute
505 King Avenue
001 Columbus, OH 43201

709 Plastics Tech Eval Center
Picatinny Arsenal, Bldg 176
Attn: Mr. A. M. Anzalone
001 Dover, NJ 07801

710 KETRON, INC.
Attn: Mr. Frederick Leuppert
1400 Wilson Blvd
Architect Bldg
002 Arlington, VA 22209

711 Metals and Ceramics Inf Center
Battelle
505 King Avenue
001 Columbus, OH 43201

Commander
US Army Test & Evaluation Command
ATTN: Mr. J. Weintraub
001 Aberdeen Proving Ground, MD 21005

Commander
US Army Security Agency
Attn: IARDA-CGS
001 Arlington Hall Station, VA 22212

Mr. Seymour Jeffery
Room A247, Bldg 225
National Bureau of Standards
001 Washington, DC 20234

Project Manager, MSCS
Attn: DRCPM-MSCS-LMD
001 Fort Monmouth, NJ 07703

Commander
US Army Electronics Command
Attn: DRCPM-ATSS
001 Fort Monmouth, NJ 07703

Commander
Central DA TMDE Activity
Attn: DRXMD-TR
001 Lexington, KY 40507

Commander
Sacramento Army Depot
Attn: DRXSA-MPE-3
001 Sacramento, CA 95813

DISTRIBUTION LIST (Continued)

Commander
US Army Missile Command
Attn: DRSMI-RA
001 Redstone Arsenal, AL 35809

Commander
US Army Tank Automotive Command
Attn: DRDTA-REA
001 Warren, MI 48090

Joint Tactical Communications (TRI-TAC) Office
Attn: TT-LD-ILS
001 Fort Monmouth, NJ 07703

Headquarters Marine Corps
CODE LMC-4
001 Washington, DC 20380

Commander
Frankford Arsenal
Attn: SARFA-FCF
001 Philadelphia, PA 19137

Director, National Security Agency
NSA/S274
Attn: Mr. Ron Seaman
001 Ft. Meade, MD 20755

Commander
Tobyhanna Army Depot
Attn: DRXTO-MI-O (Mr. J. Molenda)
001 Tobyhanna, PA 18466

Commander
Picatinny Arsenal
Attn: Mr. L. Goldsmith
TSD Bldg 352
001 Dover, NJ 07081

Project Manager, ARTADS
Attn: DRCPM-TDS
001 Fort Monmouth, NJ 07703

Project Manager, MALOR
Attn: DRCPM-MALR
001 Fort Monmouth, NJ 07703

Project Manager, SINCGARS
Attn: DRCPM-GARS
001 Fort Monmouth, NJ 07703

Chief, Special Items Management Office
Attn: DRSEL-SI
001 Fort Monmouth, NJ 07703

Commander
US Army Materiel Systems Analysis Activity
Attn: DRXSY-CC
001 Aberdeen Proving Ground, MD 21005

Project Manager, ATACS
Attn: DRCPM-ATC
001 Fort Monmouth, NJ 07703

DISTRIBUTION LIST (Continued)

Project Manager, NAVCON
Attn: DRCPM-NC
001 Fort Monmouth, NJ 07703

Project Manager, SIGINT/EW
Attn: DRCPM-SIEW
001 Fort Monmouth, NJ 07703

Commander
US Army Communications
Systems Agency
Attn: CCM
001 Fort Monmouth, NJ 07703

DATE
FILMED

-7